NPS ARCHIVE
1997
PURCELL, T.

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

THESIS

THE USE OF CLASSIFICATION TREES TO CHARACTERIZE THE ATTRITION PROCESS FOR ARMY MANPOWER MODELS

by

Terence S. Purcell

September, 1997

Thesis Advisor: Robert R. Read
Second Reader: Samuel E. Buttrey

Thesis
P9482

Approved for public release; distribution is unlimited.

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE September 1997

3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE The Use of Classification Trees to Characterize the Attrition Process for Army Manpower Models

5. FUNDING NUMBERS

6. AUTHOR(S) Terence S. Purcell

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. **ABSTRACT** ***(maximum 200 words)***

The U.S. Army has a system of large personnel-flow models to manage the soldiers. The partitioning of the soldiers into groups having common behavior is an important aspect of such models. This thesis presents Breiman's Classification and Regression Trees (CART) as a method of studying partitions relative to loss behavior. It demonstrates that CART is a simple technique to use and understand while at the same time still being a powerful forecasting tool. A CART example is included that provides the reader a thorough understanding of the method. The analysis explores the structure found in the current Classification Groups (C-Groups) used by the Army. CART is used to review the structure of the C-Groups and conduct some exploratory work to demonstrate that different combinations of factors result in greater internal homogeneity in forecasting. Recommendations are provided on how to approach the process of modifying the C-Groups. The use of CART results in obtaining insights into the Army force structure that would not have been found with any other forecasting technique. This thesis reveals the power of CART as a forecasting tool.

14. SUBJECT TERMS Loss Rates, CART, Partition, Attributes, Cross-Validation, Tree

15. NUMBER OF PAGES 96

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

Approved for public release; distribution is unlimited.

THE USE OF CLASSIFICATION TREES TO CHARACTERIZE THE ATTRITION PROCESS FOR ARMY MANPOWER MODELS

Terence S. Purcell
Lieutenant Commander, United States Navy
B.S., Pennsylvania State University, 1981

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1997

NPS ARCHIVE
1997.09
PURCELL, T.

~~Thesis~~
~~P9482~~
~~C.2~~

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

ABSTRACT

The U.S. Army has a system of large personnel-flow models to manage the soldiers. The partitioning of the soldiers into groups having common behavior is an important aspect of such models. This thesis presents Breiman's Classification and Regression Trees (CART) as a method of studying partitions relative to loss behavior. It demonstrates that CART is a simple technique to use and understand while at the same time still being a powerful forecasting tool. A CART example is included that provides the reader a thorough understanding of the method. The analysis explores the structure found in the current Classification Groups (C-Groups) used by the Army. CART is used to review the structure of the C-Groups and conduct some exploratory work to demonstrate that different combinations of factors result in greater internal homogeneity in forecasting. Recommendations are provided on how to approach the process of modifying the C-Groups. The use of CART results in obtaining insights into the Army force structure that would not have been found with any other forecasting technique. This thesis reveals the power of CART as a forecasting tool.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

The author would like to thank Mr. Dennis Mar and Ms. Helen Davis from Academic and Administrative Computing Services at the Naval Postgraduate School for their assistance with preliminary manipulation of data. Also, many thanks to Debbi Kreider and Alan Pires from Glasgow Computer Support Group at the Naval Postgraduate School for their assistance with computer hardware and software.

EXECUTIVE SUMMARY

The U.S. Army has a system of large personnel-flow models to manage the soldiers. The partitioning of the soldiers into groups having common behavior is an important aspect of such models. This thesis presents Breiman's Classification and Regression Trees (CART) as a method of studying partitions relative to loss behavior. The ability to understand how various combinations of factor levels can produce stable levels of attrition behavior is useful for various aspects of planning and the preparation of more finely tuned loss rate forecasting.

The source of the data used for this thesis is the Small Tracking File (STF). The STF is part of the data base that supports the Enlisted Loss Inventory Model - Computations of Manpower Programs. The STF contains demographic information and gain/loss history on every non-prior service enlisted soldier. We used a six year period, January 1983 to December 1988. Only first-term enlistees are studied.

This thesis demonstrates that CART is a simple technique to use and understand while at the same time still being a powerful forecasting tool. A brief example is included which introduces the reader to the features of CART. Resource limitations required that the data be merged over time, the use of attributes be selective and that sampling be used. Four behavior categories are used: two are pre-contract term losses (adverse and non-adverse), one a full term loss, and one an extension or re-enlistment. The factors used to classify or forecast these behavior categories are Education Group, AFQT score, Gender,

and contract Term. These factors are used with coarse and non-coarse partitionings into factor levels, leading to two separate studies. These two studies are performed twice each, once without race as a factor and once with race.

For each study, classification trees are grown to about a dozen terminal nodes. These nodes produce the rates of classification for the four behavior categories and the number of soldiers included in the nodes. The different combinations of factors result in greater internal homogeneity in forecasting. The use of CART results in obtaining insights into the Army force structure that would not have been found with any other forecasting technique. The current Army practice uses only the four basic factors to classify soldiers and predict loss behavior. The details are compared with those produced by the trees.

The addition of the fifth factor (race) resulted in that factor becoming the most important one. Perhaps the most conspicuous result is that the re-enlistment rates among blacks is typically 40% or more. This rate is seldom above 30% for other groups. Exploring the factors to determine their importance in predicting loss behavior is easily conducted in CART. When a factor has little predictive power, CART will not use the factor. This is an advantage over other forecasting techniques where all factors included in the model must be used.

There is a need to seek additional explanatory factors and variables. The use of CART ensures that only variables with a high value of predictability will be included. CART is an uncomplicated method. Once the techniques are learned, they are simple to use and easy to explain. This thesis reveals the power of CART as a forecasting tool.

I. INTRODUCTION

A. BACKGROUND

Manpower planning is often defined as the attempt to match the supply of people with the jobs available for them.(Bartholomew, Forbes, and McClean, 1991) Performing manpower planning is an important function at any organization. The importance of this function increases as the size of the organization increases. The number of people required and the number of people available are the two features of most manpower planning problems that must be addressed.

The U.S. Army is like any other organization when it comes to manpower planning. The function must be performed. The Army is complicated by its enormous size. What are the manpower requirements of the Army and how many people should the Army recruit to meet its requirements? In order to answer these questions, future actions of the Army's current soldiers must be forecasted. The Army must identify groups of soldiers with homogeneous attributes that share common behaviors.

When a soldier first enlists in the Army, that soldier is referred to as a "first-term enlistee" and the soldier enters into a contract to remain in the Army for a specified time. This specified time is referred to as the soldier's "term of enlistment" in the Army or "commitment" to the Army. Forecasting what a first-term enlistee will do is an important function in the Army's manpower planning This group of soldiers account for a large portion of the Army's enlisted personnel. Will the enlistee stay in the Army past his or her commitment, will the enlistee complete his or her commitment and exit the Army, or will the enlistee be separated from the Army prior to completing his or her commitment? Successful manpower planning depends on the ability to describe and predict patterns of loss.

1. The U.S. Army's System

One component of the Army's computer based models developed in the early 1970's to meet demands for improvement in manpower planning and budgeting is called the Enlisted Loss Inventory Model - Computations of Manpower Programs (ELIM).* *Loss rates* are the important parameters estimated by ELIM. These loss rates are used in within ELIM and other Army models to further develop the manpower plan. The reliability of the manpower plan is determined by the accuracy of these forecasted loss rates.

Loss rates attempt to forecast the proportion of soldiers that will leave the Army. A loss rate is simply the number of people who left the Army, divided by the total number in the Army. Loss rates are constructed from historical data, analyzed, and forecasted into the future. Separate loss rates for sub-populations of soldiers are developed by the Army. For example, first-term enlistees are grouped together in cohorts. A cohort simply defines when, by year and month, a group entered the Army. A loss rate could be developed for each cohort. Cohorts could be broken down by other data characteristics and additional loss rates could be constructed. Constructing loss rates in this fashion for portions of the Army's first-term enlistees led to the development of Classification Groups.

* The model's designated acronym is ELIM-COMP. However, ELIM has become the most widely accepted way of referring to the model and will be used throughout this thesis.

2. Classification Groups

Soldiers that have never been in the service prior to their current enlistment are referred to as non-prior service (NPS) soldiers. First term enlistees that are NPS are partitioned into one of ten Classification Groups (C-Groups). Each C-Group is determined by a soldier's gender, education, Armed Forces Qualification Test (AFQT) score, term of enlistment, and entry level training time. The current C-Groups are presented in Table 1.1.

Once soldiers are place into C-Groups, ELIM uses this information to forecast first term loss rates based on historical loss activity. Upon completion of forecasting loss rates, ELIM will go onto project forecast force strength in any time period.

B. THESIS OBJECTIVES AND ORGANIZATION

1. Objectives

ELIM uses exponential smoothing to forecast loss rates. CAPT E. T. DeWald, USMC, wrote a Master's Thesis (DeWald 1996) that explored several other Time Series methods of forecasting loss rates. When DeWald partitioned his data into C-Groups, he found that 45 percent of the active Army accessions were from C-Group 1. DeWald concluded that the utilization of any loss rate forecasting technique would have to be accurate with respect to C-Group 1 or it would not be accepted. Therefore, DeWald only considered C-Group 1 in his thesis work. DeWald concluded that the exponential smoothing method currently employed by ELIM was a valid way of calculating the forecasts. Building from DeWald's data base, this thesis has two objectives.

C-Group	Gender	Education	AFQT Category	Term
1	M	HSD	I - IIIA	3, 3VEL, 4, 4VEL
2	M	HSD	IIIB	3, 3VEL, 4, 4VEL
3	M	HSD	IV - V	3, 3VEL, 4, 4VEL
4	M	NoHSD	I - IIIA	3, 3VEL, 4, 4VEL
5	M	NoHSD	IIIB - V	3, 3VEL, 4, 4VEL
6	F	HSD	I - IIIA	3, 3VEL, 4, 4VEL
7	F	HSD	IIIB - V	3, 3VEL, 4, 4VEL
8	F	NoHSD	I - V	3, 3VEL, 4, 4VEL
9	M	HSD & NoHSD	I - V	2, 2VEL, 5, 6
10	F	HSD & NoHSD	I - V	2, 2VEL, 5, 6

TABLE KEY

C-Group: Characteristic Group Number
Gender: M -> Male
F -> Female
Education: HSD -> High School Degree
NoHSD-> No High School Degree
(Actual acronyms used in ELIM are HSDG and NHSDG)

AFQT Cat:		
	I-IIIA	-> 50 to 99 percentile
	IIIB	-> 31 to 49 percentile
	IV	-> 20 to 30 percentile
	V	-> 0 to 20 percentile

Term: Length of Enlistment Contract (in Years) VEL indicates a Variable Enlistment Length contract. The length of enlistment begins at the completion of training.

Table 1.1 Currently Defined Characteristic Groups

The first objective of this thesis is to review the structure of the current Army C-Groups and conduct some exploratory work in an attempt to demonstrate that different combinations of factors can produce greater internal homogeneity in forecasting. Classification and Regression Trees (CART) (Breiman, Friedman, Olshen, and Stone, 1984) is presented as a method of completing the tasks associated with accomplishing this objective. It is hypothesized that including new factors and / or excluding old factors will provide a more accurate way of defining C-Groups. The Army manpower models have the ability to forecast month by month. CART is most useful in long term planning .

The second objective of this thesis is to present a method of forecasting loss rates to support high level administrative decisions. CART will be used to forecast loss rates. It differs from time series methods and can be less complicated to conduct and more easily understood.

2. Organization

The background and objectives of this thesis have been provided in this introduction. In Chapter II the reader will learn about the data used in this thesis including the source of the data, its contents, and what had to be done to the data to make it useful. Chapter III provides a description of the methodology. This chapter includes an introduction to CART, a description of CART used in S-Plus, and a CART example. The analysis of the data and the results obtained from the analysis are provided Chapter IV. Conclusions and recommendations are provided in Chapter V.

It is important to note at the outset that the scope of this thesis was limited by the data file. Due to the size of the file, it was not feasible to work with the entire data file.

Sufficient resources for a more thorough study were not available. It was necessary to perform the analysis on a representative sample of the original data file. The origin of the sample file and how is was derived is discussed in Chapter II.

II. DATA EVOLUTION

There were significant challenges and hurdles to cross involving the data before it could be used in this thesis. The issues encountered and resolved concerning the data are deep, and warrant a large resource commitment. Their importance is to be emphasized.

A. SOURCE

Much of the documentation for the ELIM system is produced by the General Research Corporation (GRC). The GRC documentation provides a detailed description of the modules and files in the ELIM system (GRC, 1989). For example, the Small Tracking File (STF) of ELIM contains demographic information and gain/loss history on every non-prior service enlisted soldier that joined the Army during a six year period. The source of the data used for this thesis is the STF for the period from January 1983 (cohort 8301) to December 1988 (cohort 8812). The information contained in the STF comes from the two other Army files, the Enlisted Master File (EML) and the Gain/Loss Transaction File (GLF). Monthly extracts are taken from EML and GLF and merged together to form STF.

B. FORMAT AND MANIPULATION

The data was collected, prepared, and stored as SAS system files on IBM 3480 tape cartridges by GRC. Using SAS, the data on the cartridges was copied to a 3390 disk, which was attached to an Amdahl 5995 running the IBM MVS/ESA operating system at the Naval Postgraduate School (NPS). The data file residing on the mainframe computer accounted for over 722,745 soldiers, one line of data for each soldier. The data contained social security numbers and other information for each soldier that would not be necessary. Using SAS, a new file was created by removing the information that was not necessary from the original file. Table 2.1 contains the information and format for the new file. This file was then converted to a flat file and FTP'd to a UNIX account.

Character Location	Contents
1-4	Cohort (YYMM)
6-7	AFQT Percentile Score
9	Race (Numeric Code)
11	Gender (M or F)
13	Length of Term of Service (in Years)
15	Civilian Education Level (Alpha Code)
17-19	Age at Entry (in months)
21-24	End of Term of Service Date (YYMM)
26	Service Component (i.e. R)
28-29	Current Training Time (MM)
31	VEL Flag
33-34	Number of Events (max used was 13)
36-37	# of Mths From Cohort When Event Took Place
.	
.	(continues for 13 events)
.	
72-73	# of Mths From Cohort for 13th Event
75-77	Loss/Gain Event Code(Alpha Code)
.	
.	(continues for 13 events)
.	
123-125	Loss/Gain Event Code for 13th Event

Table 2.1 Information and Format of UNIX File

The analysis of the data was performed in S-Plus on a 486/166 personal computer (PC). The size of the file remaining in the UNIX account prohibited its use in S-Plus on a PC. The C programming language was used to manipulate the file while it was in the UNIX system, attempting to reduce the size of the file so it could be used in S-Plus. The file consisted of loss/gain information for each soldier and attributes for each soldier. The attributes for each soldier included Cohort, AFQT percentile score, race, gender, age, length of term of enlistment, a Variable Enlistment Length (VEL) code, and a code for the civilian education level achieved. Since data were abundant, it was decided that if an attribute in a line of data did not have a entry or if the entry was in error, that line of data would be removed. Also, any lines of data with a VEL code present were removed. Soldiers with a VEL code represented less than 3% of the data file and a separate analysis would have had to be performed if they were included in the data set.

The file that remained contained the information necessary to determine when a soldier was considered a loss to the Army. Calculating when a soldier became a loss to the Army was vital to the analysis of the data. The loss/gain codes in the file were used to perform the calculation. One or more of the loss/gain codes were present in each line of data file. The file became the data source for a C program that scanned the loss/gain codes for each soldier. Each line of data was assigned to one of four "Loss" type categories. The program would assign a soldier to the "Early Adverse" (Eadv) category if that soldier was released from the Army for an adverse reason prior to the end of his/her obligated term of service. Other soldiers who were released early from the Army were assigned to the "Early Okay" (EOK) category if the reason for their release was not under adverse conditions. If a soldier was discharged at the end of his/her obligated term of enlistment, that soldier was placed in the "End of Term" (EndT) category. Finally, if a soldier remained in the Army past the end of his/her first term of obligated service, that soldier was place in the "Not Lost" (Not) category. The program made these assignments based on the loss/gain code scanned. When the program scans the codes, gain codes are ignored. Lines of data are assigned to a Loss category based on the first loss code found during the scan. If the program finds no loss codes (a line of data contained only gain codes), that data line would be assigned to the "Not Lost" category. If the first loss codes found was EXT or IMR, that line of data was also assigned to the "Not Lost" category. All the loss codes were assigned to one of the four Loss categories. The loss/gain codes, their definitions, and the Loss category they were assigned to, can be found in Table 2.2.

Once the Loss type was determined for a line of data, it was added to the data. The loss/gain codes and attributes that were not needed were then removed from the data file. Finally, the AFQT percentile scores and Civilian Education Codes were placed into categories. The AFQT categories can be found in Table 1.1. The normal educational groupings (EdGrp) associated with the Civilian Education codes are "No High School Degree" (NoHSD) and "High School Degree" (HSD). For this study, the categories "General Education Development" (GED), "two or less years of college" (<=2YrsColl), and "more than two years of college" (>2YrsColl), were added to the educational groupings.

As a result of all the these actions, the file now accounted for 687,212 soldiers and required 22 megabytes of disk storage space. Although this file could be read into S-Plus,

----------LOSS CODES----------

		LOSS
CODE	DEFINITION	CATEGORY
DFR	Dropped From Rolls	EAdv
EDP	Expeditious Discharge Program (UNSAT Performance)	EAdv
MCD	Msiconduct Discharge	EAdv
TDP	Trainee Discharge Program	EAdv
UFT	Unfit For Duty	EAdv
ERL	Early Release	EOK
HRD	Hardship Discharge	EOK
MPP	Marriage/Pregnancy/Parenthood/Dependency	EOK
LLL	Unknown Loss Type	EOK
OTH	Other - weight control, erroneous entry, etc.	EOK
PHY	Physical Disability	EOK
RET	Retirement	EOK
SCH	School	EOK
ETS	Expiration of Term of Service	EndT
OSR	Overseas Returnee	EndT
EXT	Extension	Not
IMR	Immediate Reenlistment	Not

-----GAIN CODES-----

CODE	DEFINITION
G90	Greater Than 90 Day Reenlistment
L90	Less Than 90 Day Reenlistment
NPA	No Prior Army, Army Reserve, National Guard Service
NPG	No Prior Service in Any Service, Reserve, or Guard
OTG	Other Gains - Former Officer, Warrant Officer, Admin Error, etc.
RMC	Return to Military Control
RSV	Gain From National Guard or Reserves

Table 2.2 Loss/Gain Codes

CART analysis could not be performed on the data. A smaller file was created by splitting the STF file in half, only including data from January 1986 (cohort 8601) to December 1988 (cohort 8812). This smaller file accounted for 329,762 soldiers and required 10 megabytes of disk storage space. CART analysis could be performed on this file using S-Plus, but required overnight processing. This was considered unreasonable so a 10% random sample was taken from the smaller file. The 10% sample required only 1 megabyte of disk storage space and CART analysis was easily performed in S-Plus with the sample file. Summary statistics were collected from the sample file and the entire data set and can be found in Appendix A. These statistics included the percentage of each attribute and Loss category found in the data sets. The goal was to show that the sample file was an accurate representation of the characteristics found in the entire data set. All the statistics gathered from the sample file were within plus or minus two percentage points of the statistics collected from the entire date set. For example, it was determined that the entire data set consisted of 13% females and 87% males while the sample data consisted of 14% females and 86% males. The conclusion was that the sample file was an accurate representation of the characteristics of the entire data file and analysis performed on the sample file would mirror analysis performed on the entire data file. Appendix B contains the first 40 rows of the sample file.

III. METHODOLOGY

A. INTRODUCTION TO CART

Tree-based models are a non-parametric technique used in statistics to uncover structure in a data set. These type of models can be used in both regression and classification-type problems. Regression and classification models attempt to predict the value of the dependent variable based on the value of a set of independent variables. The difference between regression and classification models is the type of dependent variable involved. Regression-type problems have a continuous dependent variable, while classification-type problems have a dependent variable that is categorical. When using tree-based models, if the dependent variable is continuous, the tree that is grown is called a regression tree. Likewise, if the dependent variable is categorical, the tree that is grown is called a classification tree. Since this thesis constructs and analyzes classification trees, this introduction and subsequent example will focus on classification trees.

Breiman *et al.* (1984) introduced tree-based models to the mainstream statistical audience and they developed the computer program CART (Classification and Regression Trees). CART has since become a generic term that refers to the use of a tree-based regression and classification scheme that identifies the important variables and is free of linearity constraints. CART offers an alternative to the linear logistic and additive logistic models used for classification. According to Chambers and Hastie (1992), the use of tree-based models is in its infancy but the method is gaining widespread popularity as a means of devising prediction rules for rapid and repeated evaluation, as a screening method for variables, as a diagnostic technique to assess the adequacy of linear models, and simply for summarizing large multivariate data sets. CART has several advantages over more familiar classification techniques that makes it particularly attractive. CART is more easily interpreted, it has the ability to handle multiple responses, and it is also capable of handling a mix of categorical and continuous independent variables.

An understanding of tree terminology is required to understand CART. A *tree* is a collection of nodes that are connected together. The node at the top of the tree is called the *root* node. If node y is below and directly connected to node x, then y is said to be a *child* of x, and x the *parent* of y. The root node in a tree is the only node without a parent. The nodes at the bottom of a tree have no children. Each of these nodes is called a *leaf* or *terminal* node. Nodes other than the root node or terminal nodes are called *interior* nodes. The *depth* of any node in a tree is the length of the unique path from the root node to the node in question. Thus, the root node has depth 0 and the child nodes of the root node have depth 1. (Weiss, 1995)

A binary tree is a tree in which no node can have more than two children. CART is so named because the primary method used to display the results of the analysis is in the form of a binary tree. In order to predict the dependent variable from the set of independent variables, one follows a path from the root node, through the interior nodes, to the terminal nodes of the tree. At the root node and each interior node encountered, a choice must be made to go to the left child node or the right child node according to some "best" splitting criterion. CART is an iterative procedure that attempts to separate all the cases of a data set into nodes of a binary tree that are "homogenous" or "pure." The splitting criterion implemented determines the purity of a tree.

In CART, each data point is called a "case" and each case falls into one of several "classes" (indexed by k). The root node contains all the cases in the data set. Splitting the data set at the root node involves examining every possible split of the cases and picking the split that gives the greatest increase in purity. The tree algorithm searches through M independent variables $\left(x_1, x_2, \ldots, x_M\right)$ one by one, and evaluates the change in purity. The "best" split will be at a specific value, j, of a single independent variable, x_m. If the split is on a numeric independent variable, all cases for which $x_m < j$ will be placed in the left child node and all cases for which $x_m \geq j$ will be placed in the right child node. For example, if the independent variable is age (measured in years), and $j = 22$, then the left child node will include all ages below 22. The right child node will include all ages 22 and above. If the split is on a categorical independent variable, the left child node will receive a portion of the entire group. If the independent variable was gender (male and female), and j = male, all females would be placed in the left child node and the right child node would receive all males.

B. S-PLUS AND CART

The criteria used to split the data in S-Plus differs slightly from the recursive partitioning methods used in Breiman *et al* (1984). S-Plus uses the deviance (likelihood ratio statistic) to measure the purity. The smaller the deviance, the greater the purity.

Impurity, or deviance, is measured at every node. The total deviance of the tree is the sum of the deviances at the terminal nodes.

The model used in S-Plus for classification is based on the multinomial distribution, with parameter u_i, where i designates the node in the tree. The vector $u_i = (p_1, p_2, \dots, p_k)$, such that $\sum_k p_k = 1$, is the probability distribution over the k classes at node i. At each node i, n_{ik} cases are observed in class k, where $\sum_k n_{ik} = n_i$ (the total number of cases at node i). The deviance function at a node is defined as minus twice the log-likelihood,

$$D_i = -2\sum_k n_{ik} \log p_{ik} \ . \tag{3.1}$$

For node i, an estimate for u_i must be made because the probabilities are unknown. Such an estimate would be

$$\hat{u}_i = \left(\hat{p}_{i1}, \hat{p}_{i2}, \dots, \hat{p}_{ik}\right) \quad \text{and} \quad \hat{p}_{ik} = \frac{n_{ik}}{n_i}, \text{ for all } k \, .$$

Thus, the deviance function used in S-Plus becomes

$$D_i = -2\sum_k n_{ik} \log \hat{p}_{ik} \ . \tag{3.2}$$

The split that results in the greatest increase in purity is the split that maximizes the change in deviance (goodness-of-split). The change in deviance is the deviance of the node i minus the deviance of the left child node (l) minus the deviance of the right child node (r). Symbolically, the change in deviance that one would want to maximize is expressed as

$$\Delta D_i = D_i - D_l - D_r \, . \tag{3.3}$$

A single terminal node is said to be 100% pure (deviance =0) if all the cases in that node are of the same class. If the tree is grown without constraints, a tree can have as many terminal nodes as there are observations. A tree of this type characterizes the structure of the data perfectly and would have zero total deviance. This situation may be likened to that of representing *n* points in the plane with a polynomial of degree *n*-1. The fit may be perfect, every residual equals zero, but there is no credibility to its usefulness for prediction. A tree with zero deviance may well be worthless for predicting the classification of data not found in the data used to grow the tree.

S-Plus uses one of two stopping criteria to decide whether to split a node and ensure that a tree is not grown to 100% purity. A split will not occur at a node if the node deviance is less than some pre-determined value or if the number of cases in a node is smaller than some pre-chosen minimum. The default values in S-Plus are 0.01 and 10, respectively.

A tree's size is measured by its number of terminal nodes. Even with the stopping criteria in place, a tree may be grown to a size that is beyond that which can be useful. A tree such as this is called an "overgrown" tree. Creating an overgrown tree from a data set is done by design so that the growth of the tree will uncover all relevant structure in the data. Once the entire structure is uncovered, the tree is then "pruned" back to a useful size. In S-Plus, the methods of pruning and cross-validation are closely related. Both methods will be examined more closely. Pruning compares tree size with deviance. Once a tree size is determined, this information can be provided to the pruning method. When S-Plus executes the pruning method, it recursively snips off the least important splits until the tree is the size specified. Cross-validation is a technique that is used to assist in the selection of the optimal tree size, a size that optimizes both the purity of the tree and its ability to predict from new data.

An overgrown tree that was created from the entire data file, is used as the input to the pruning method. The pruning method in S-Plus can be executed in one of two ways. When a tree size is also part of the input, S-Plus will grow the pruned tree to the specified size using the nodes that achieve the lowest deviance. If a tree size is not specified, S-Plus will determine a nested sequence of subtrees by recursively snipping off the least important splits of the tree provided. The subtrees will span a range of tree sizes. When the pruning method is executed without a tree size provided and then plotted, a plot of the range versus deviance is made available. The CART example in this chapter executes the pruning method in both ways and will provide the opportunity to visually examine a pruning plot.

Using the deviances of any tree, as a measure of the tree's predictive ability, leads to an overly optimistic choice because the deviances are based on the same data used to construct the tree. The technique of cross-validation is a way to counter this problem. It exploits the use of an independent sample to assess the predictive ability of a tree. The cross-validation (CV) method, supplied in the software, divides the data into M mutually exclusive sets. Each of the M sets serves as an independent test set for trees grown on the learning sets. The learning set is composed of the union of the M-1 remaining subsets. The M mutually exclusive sets are generated at random from the data file. The number of mutually exclusive sets can be specified, but ten is the default value in S-Plus. The same overgrown tree that is used as input to the pruning method, is also used as input to the cross-validation (CV) method. The first execution of the pruning method within the CV method is done by providing only the overgrown tree as input. No tree size is provided to the pruning method. The CV method then records the range of tree sizes generated by the pruning method. In 10-fold cross validation, each of the 10 sets is held out in turn and a "learning" tree is grown to the remaining nine sets.* The CV method then executes the pruning method once again. Each time the pruning method is executed at this stage of the CV method, the pruning method will utilize three input parameters. The first input parameter is the "learning" tree grown from nine tenths of the data. The second parameter is the range of tree sizes generated by the first execution of the pruning method. No tree size is specified during this execution of the pruning method. By providing the range, the nested sequence of subtrees will be created over the same range as the sequence created during the first execution of the pruning method. The third parameter, called *newdata*, is the remaining one tenth of the data that was held out when the "learning" tree was grown. *Newdata* is used to evaluate the nested sequence of subtrees. Using equation 3.2, deviances are accumulated from each of the 10 sets based on the misclassification rate of *newdata*. The $\hat{p}_{ik}$'s for the equation are generated when the "learning" tree is grown from nine tenths of the data. The n_{ik}'s are taken from *newdata* and form the one tenth of the data held back. When executed in S-Plus, this procedure will create an object of class "tree.sequence" and can be plotted. A cross-validation plot is a plot of the range (i.e., tree

* If the stopping criteria were removed to create the overgrown tree, they must also be removed to create the "learning" trees in the cross-validation method. Appendix C provides details.

size) versus the total accumulated deviance. The CART example in this chapter demonstrates the execution of the CV method. The example will also provide an opportunity to visually examine a cross-validation plot.

The node numbering pattern used by S-Plus must be understood in order to examine a tree plot. The root node of a binary tree is numbered 1. The left child node is numbered 2 and the right child node is numbered 3. Each level is numbered from left to right. Figure 3.1 is a *full* binary tree of depth three that displays this numbering pattern. The binary tree is *full* because each node, except the terminal nodes, has exactly two child nodes and each level is full. When growing trees, S-Plus will always grow a full binary tree in order to number the nodes. After the nodes have been numbered, S-Plus will examine the interior nodes to see if they should have been split. If an interior node should not have been split, S-Plus trims off any portion of the tree below the node being examined. However, the numbering of the nodes is not adjusted for this; it will remain the same. For example, suppose node five in Figure 3.1 should not have been split due to one of the stopping criteria. Figure 3.2 is the result of S-Plus growing the exact same tree as that found in Figure 3.1, numbering the nodes, and then trimming off nodes 10 and 11.

C. CART EXAMPLE

The discussion of CART will be furthered by means of introducing an example. A random sample of size 50 was taken from the actual data to form a small data set for this example. The example is a simplified version of the analysis performed with the actual data. Discussion of this analysis will follow the example. By following the example closely, the reader will have a more complete understanding of the procedures involved

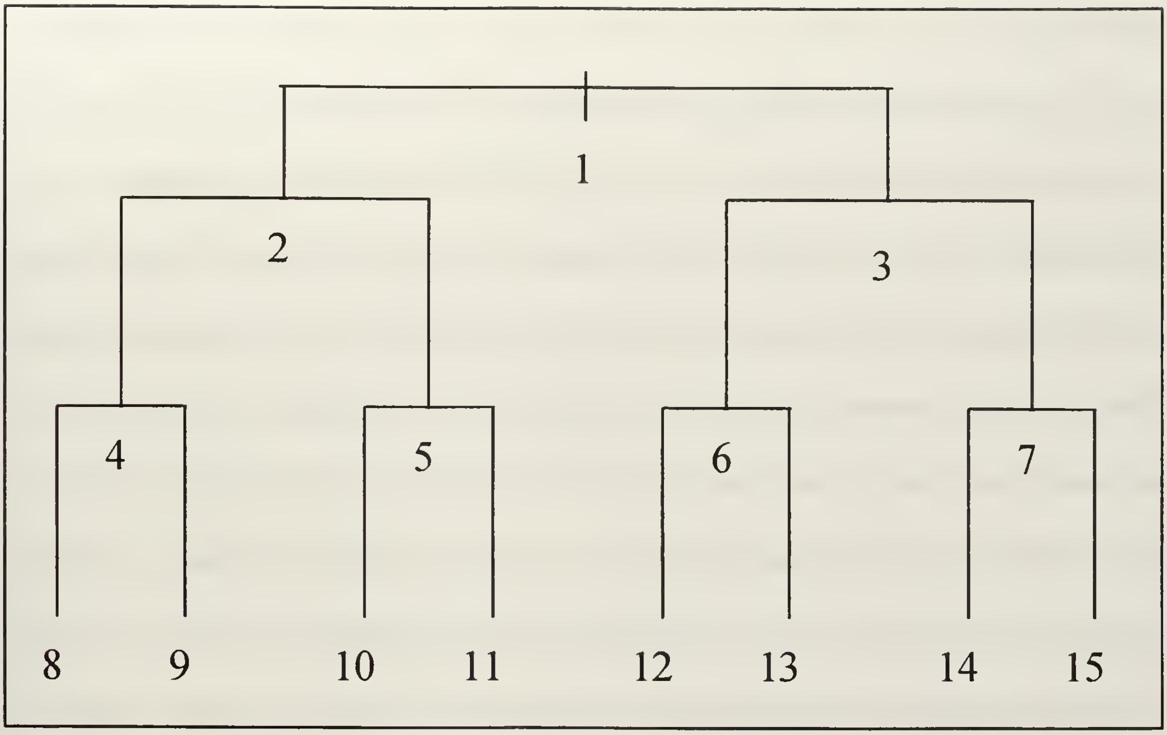


Figure 3.1 Full Binary Tree With Numbered Nodes

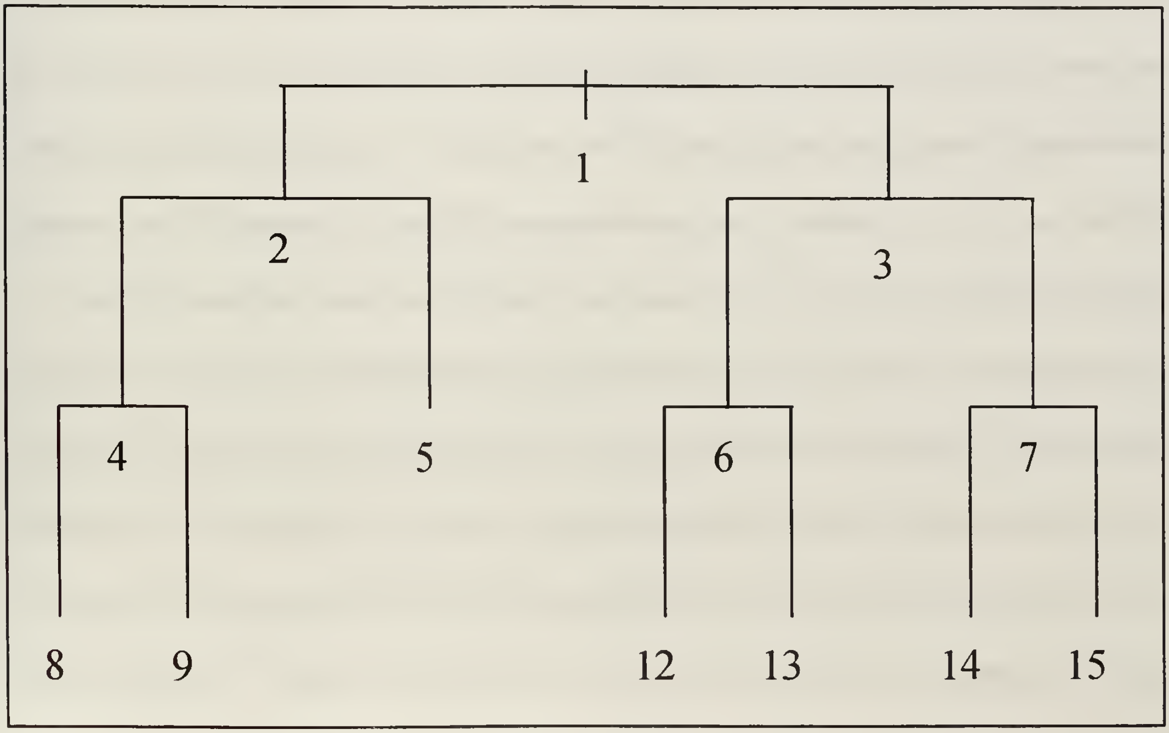


Figure 3.2 Trimmed Binary Tree With Numbered Nodes

in the CART process and executed in S-Plus. The example should enable the reader to more fully comprehend the analysis performed with the actual data.

The data for the example consists of 50 first-term Army enlistees. The independent variables, or attributes, to be used in this example are mental category based on AFQT score (AFQT); gender (Gender); length of term (Term); and education group (EdGrp). AFQT consists of six levels: I, II, IIIA, IIIB, IV, and V. Gender has two levels: male and female. Term has 5 levels: 2Yrs, 3Yrs, 4Yrs, 5Yrs, and 6Yrs. Only two levels will be used for the attribute EdGrp: HSD and NoHSD. Soldiers with any education above the high school level will be placed in the HSD level. Soldiers with only a GED will be placed in the NoHSD level. These factors are the same as those utilized in the present classification (C-Group) system in ELIM.

Each soldier represents a different case, so there are 50 cases. What happens to first-term enlistees? Soldiers either leave the Army at or before the end of their term of enlistment, or they stay in past the end of their enlistment. Our global plan is to place the soldiers into one of the four Loss type categories defined in Chapter II. But, for this example, the first three Loss type categories will be combined together to form the category "Lost." Soldiers who leave the Army at or before the end of the term of enlistment, are considered to be lost. Each soldier falls into either class *lost* or class *notlost*. Each soldier exhibits certain characteristics that the Army hopes will predict his or her likelihood of falling into one of the two classes. What percentage of soldiers is lost? What group of attributes characterizes the "typical" lost soldier? By using CART analysis in S-Plus, questions such as these can be answered. The example data and the important S-Plus commands used for this example can be found in Appendix D.

Figure 3.3 is a final classification tree grown from the example data and possessing four terminal nodes. Information not normally found on the tree has been added to the figure to emphasize some of the important features of the tree. Each node's number has been placed inside a diamond. The splitting criteria at the root node and interior nodes have been identified. Recall that terminal nodes are not split, thus their name. There are three lines of information under each node. The first line contains the deviance and number of cases (soldiers) for that node. For instance, the root node has a deviance of 59.30 and contains 50 cases. The second and third lines contain information pertaining to the *lost* and *notlost* categories. Each of these two lines contain the proportion of the total number of cases that belong in the line's category, and the actual number in that category. At the root node, the *lost* category's proportion of the total number of cases is 0.72 and the actual number of *lost* cases is 36. The root node indicates that of the 50 total cases, 72% are considered to be *lost* and 28% are considered to be *notlost*.

S-Plus examines every attribute to determine the "best" split. The "best" split selected is the one that maximizes the reduction in deviance and is made at a specific value of a single independent variable (attribute). Every possible combination of the levels within each attribute must be examined to determine the "best" split. In this example, the reader can see from Figure 3.3 that the splitting criteria at the root node is based on the attribute Term. Of the five categorical levels in the attribute Term, the splitting criteria

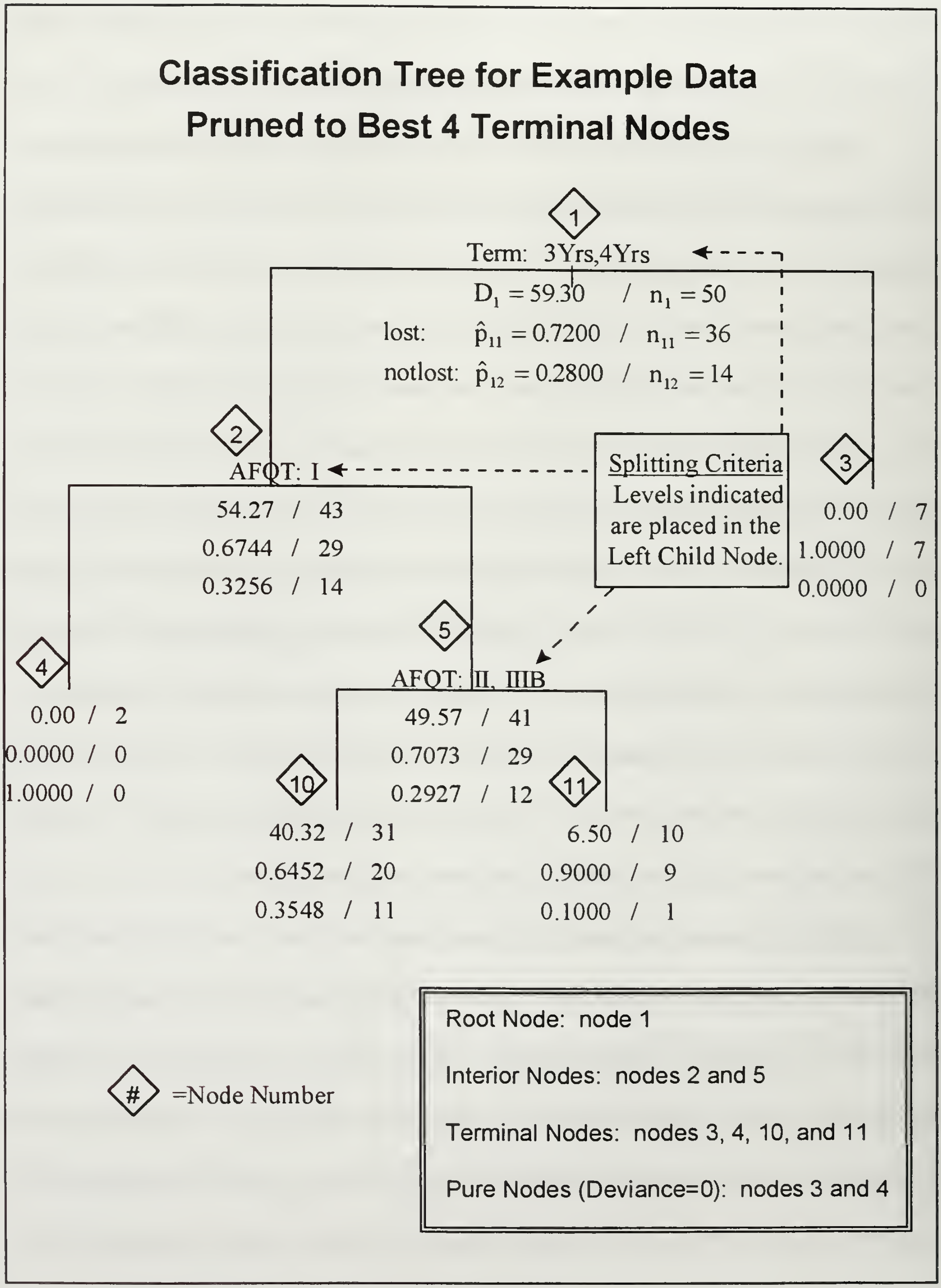

Classification Tree for Example Data
Pruned to Best 4 Terminal Nodes
1
Term: 3Yrs,4Yrs
$D_1 = 59.30$ / $n_1 = 50$
lost: $\hat{p}_{11} = 0.7200$ / $n_{11} = 36$
notlost: $\hat{p}_{12} = 0.2800$ / $n_{12} = 14$
2
AFQT: I
54.27 / 43
0.6744 / 29
0.3256 / 14
Splitting Criteria
Levels indicated are placed in the Left Child Node.
3
0.00 / 7
1.0000 / 7
0.0000 / 0
5
AFQT: II, IIIB
49.57 / 41
0.7073 / 29
0.2927 / 12
4
0.00 / 2
0.0000 / 0
1.0000 / 0
10
40.32 / 31
0.6452 / 20
0.3548 / 11
11
6.50 / 10
0.9000 / 9
0.1000 / 1
# =Node Number
Root Node: node 1
Interior Nodes: nodes 2 and 5
Terminal Nodes: nodes 3, 4, 10, and 11
Pure Nodes (Deviance=0): nodes 3 and 4

Figure 3.3

informs the reader that the left child node will contain the cases with the levels indicated in the printed split. Cases with the levels not indicated, will be placed in the right child node. Here, if the length of a soldier's enlistment term is three or four years, those cases are found in the left of the root node. If length of the enlistment is something other than three or four years (two, five, or six years), those cases are found in the right of the root node. This specific split within the attribute Term is the single split, across all predictors, that reduced the deviance by the greatest amount.

The depth of the tree at the root node is 0 and the depth at the next level below the root node is 1. The deviance of the tree at a depth of 0 is just the deviance at the root node. The deviance of the tree at a depth of 1 is the sum of deviances of the nodes in the level at this depth. The nodes at a depth of 1 are the root node's left child node (node 2) and the root node's right child node (node 3). The deviance of the tree at a depth of 1 will be lower than the deviance at the root node. To illustrate the decrease in deviance, the deviance at the root node and the two child nodes will be computed. Recall that the equation to calculate deviance is

$$D_i = -2\sum_k n_{ik} \log \hat{p}_{ik} \tag{3.4}$$

where n_{ik} is the number of cases observed in at node i in class k and $\hat{p}_{ik}$ is the estimated probability of being in class k at node i. The root node has a total of $n_1 = 50$ cases, of which $n_{11} = 36$ with class *lost* and $n_{12} = 14$ with class *notlost*. This gives $\hat{p}_{11} = \frac{36}{50} = 0.7200$ and $\hat{p}_{12} = \frac{14}{50} = 0.2800$ (numbers are printed in Figure 3.3 under the root node). Each node's deviance can be found directly under the node in Figure 3.3. The deviance of the root node (tree depth of 0) is

$$D=-2\left[36\ln\frac{36}{50}+14\ln\frac{14}{50}\right]=59.2953\,.$$

The first split in this example was made on the attribute Term. The split resulted in $n_2 = 43$ cases in the left child node (node 2) and $n_3 = 7$ cases in the right child node (node 3). When the left child node is examined, $n_{21} = 29$ cases fall into the class *lost*, while $n_{22} = 14$ cases fall into the class *notlost*. Likewise, the right child node is examined and found to have $n_{31} = 7$ cases in the *lost* class and $n_{32} = 0$ cases in the *notlost* class. The deviance of the tree at a depth of 1 is not printed in Figure 3.3 but can be found by summing the deviances of the two child nodes (nodes 2 and 3) and is calculated as

$$D=D_2+D_3=-2\left[29\ln\frac{29}{43}+14\ln\frac{14}{43}\right]-2\left[7\ln\frac{7}{7}+0\ln\frac{0}{7}\right]=54.2664+0.00=54.2664.$$

(The convention 0log0=0 is used and supported by continuity.) As previously stated and now demonstrated, this deviance is a lower value than the deviance found at the root node. This deviance is the smallest that can be achieved from examining all possible splits.

As a result of all the splitting done to construct the tree, the tree's deviance is 46.82. This number is found by adding together the deviances of all the terminal nodes. The terminal nodes are 3, 4, 10, and 11. Respectively, their deviances are 0.0, 0.0, 40.32, and 6.5. The sum of these deviances is 46.82. Notice that nodes 3 and 4 have a deviance of 0.0. Nodes 3 and 4 are considered to be "pure" nodes since no variation remains in these two nodes. The cases in these two nodes will fall into one of the two classes but not both. In node 3, all the cases (7) belong in the class *lost* while in node 4, all the cases (2) belong in the class *notlost*.

Closer inspection of a terminal node proves to be very useful. A likely terminal node to inspect would be node 10. Of all the terminal nodes, this node contains the largest number of cases. One might want to know something about the 31 cases in this node. Of the cases (soldiers) in node 10, 0.6452 belong in the class *lost* and 0.3548 belong in the class are *notlost*. The number of cases in each class can be determined from these proportions. Since there are 31 cases in this terminal, 20 (31 x 0.6452) are in the class *lost* and 11 (15 x 0.3548) are in the class *notlost*. What attributes describe the soldiers in node 10? They can be determined by tracing down the tree from the root node to node 10. The split at the root node, previously discussed, is on the attribute Term. To get to node 10, one must go left at the root node. Proceeding left at the root node includes all cases with a Term of 3Yrs and 4Yrs. Going left from the root node takes one to node 2. This node includes 43 cases. The splitting criterion at node 2 is "AFQT: I" which indicates that of the 43 cases, those containing level I of AFQT will be placed in the left child node (node 4) and all other levels of AFQT will be placed in the right child node (node 5). In order to get to node 10, one must proceed right to node 5. Node 5 contains 41 cases. These 41 cases are soldiers with a term of enlistment of 3 or 4 years and, as a result of their AFQT percentile, fall into one of mental categories II, IIIA, IIIB, IV, or V. From node 5, one must proceed to left to get to node 10. The splitting criteria at node 5, "AFQT: II, IIIB," further defines the split at node 2. Of the 41 cases at node 5, those cases with level II and IIIB of the attribute AFQT will be placed in the left child node

(node 10). The remaining cases will have an AFQT level of IIIA, IV or V and will be placed in the right child node (node 11). One arrives at node 10 by following the left split at node 5. In summary, the 31 cases in node 10 consist of those soldiers who have enlisted for three or four years and, because of their AFQT percentile, fall into either mental category II or mental category IIIB. Of these 31 cases, the proportion that belongs in the class *lost* is 0.6452 or approximately 65%. The proportion that belongs in the class *not lost* is 0.3548 or 35%.

A node is classified by the category with the largest proportion of cases. The misclassification rate of a node is the sum of the remaining proportions. In the example there are only two levels, *lost* and *notlost*. Each node is assigned one of these levels as its classification. Since there are only two levels in the example, the misclassification rate is just the proportion that corresponds to the level not assigned. In the case of node 10, it is classified as *lost* because this level has the largest proportion of cases in the node (0.6452 versus 0.3548). The misclassification rate of node 10 would be 0.3548 (11/31).

Figure 3.3 also provides the ability to predict the likelihood of a soldier being *lost* or *notlost*. By knowing a soldier's attributes, one can proceed through the tree to a terminal node. For example, suppose a new enlistee has joined the Army for 3 years and belongs in the mental category II. Using the same tracing method described above, one would arrive at node 10. This means that the new enlistee has the same attributes as the cases (soldiers) in node 10. One can estimate that the new enlistee has a 65% chance of being *lost* and a 35% chance of being *notlost*.

A fuller understanding of the CART process and its use in S-Plus can be achieved by learning how one arrives at Figure 3.3. This tree is not the original one created. The first step in the process is to build an "overgrown" tree from the data. Figure 3.4 is the created as a result of executing the tree( ) function in S-Plus with the example data and is considered an overgrown tree. The default stopping criteria were removed in order to necessary to uncover the entire structure of the data. The level of detail included is the default choice of the S-Plus system. This tree needs to be "pruned" back, but to what size? The tools are pruning and cross-validation.

Figure 3.5 is the result of executing the pruning method, without the tree size specified, and then plotting the deviance against the size. One can see that after a tree size of seven, the rate at which the variance decreases begins to decline. What size tree should be selected? One could select a tree size of 4, 6, 10, or even 14. The goal is a trade off of minimizing the number of nodes (easier to read and interpret) while not increasing deviance to an unacceptable level. Cross-validation considers the predictability of the tree and aids in the selection of the appropriate tree size.

Figure 3.6 is a plot of the ten-fold cross-validation for this example. Deviance is very small when the tree size is only one, but this is an artifact of the small amount of data being used in the cross-validation method. Discounting a tree size of one, the tree size that produces the minimum variance is one with four or five terminal nodes. The cross-validation plot indicates that a tree size of four will provide the greatest predictability. Arguments can be made for other tree sizes, but for this example, a tree size of four was

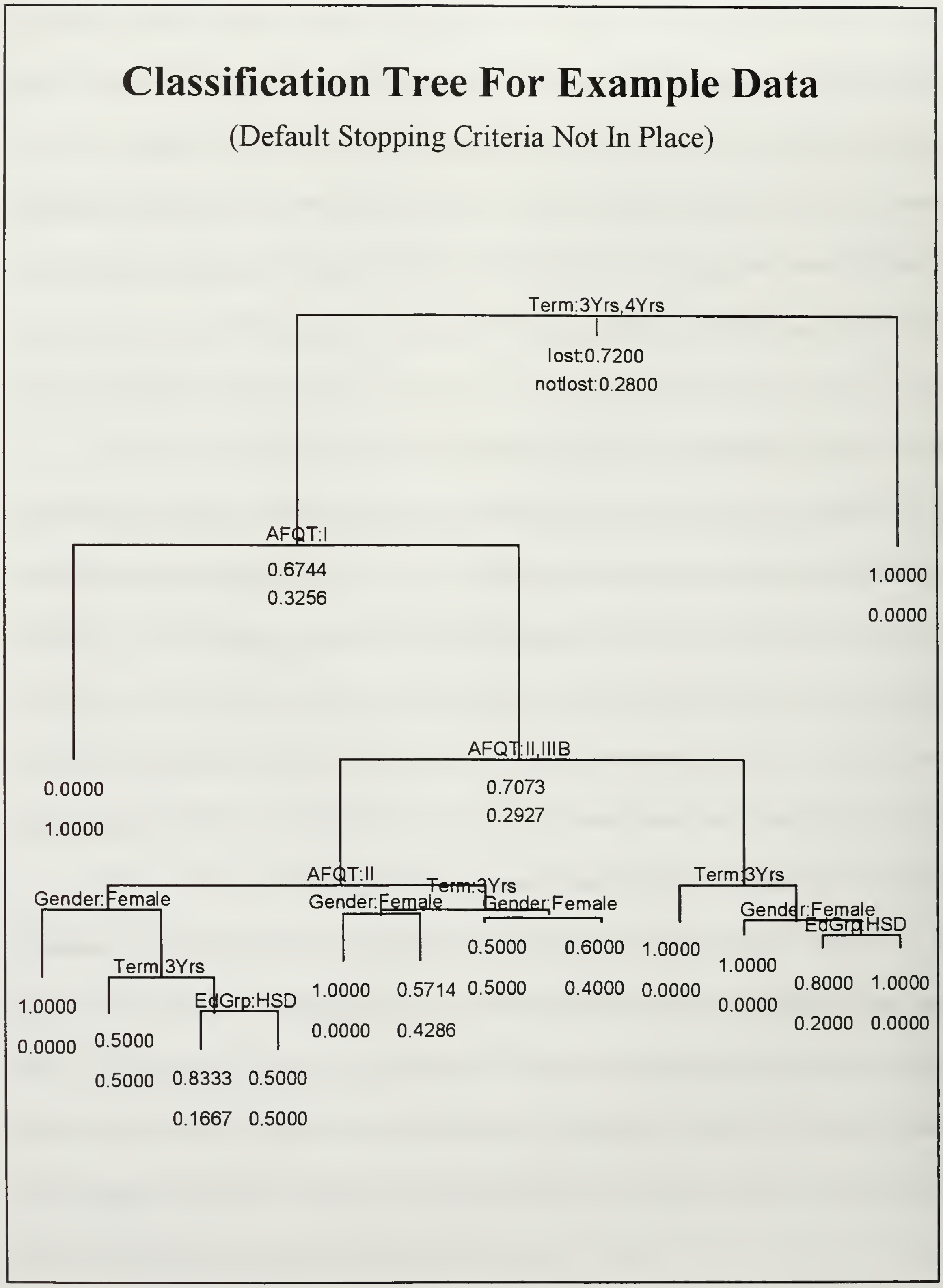

Classification Tree For Example Data
(Default Stopping Criteria Not In Place)
Term:3Yrs,4Yrs
lost:0.7200
notlost:0.2800
AFQT:I
0.6744
0.3256
1.0000
0.0000
AFQT:II,IIIB
0.7073
0.2927
0.0000
1.0000
AFQT:II
Term:3Yrs
Term:3Yrs
Gender:Female
Gender:Female
Gender:Female
Gender:Female
EdGrp:HSD
Term:3Yrs
EdGrp:HSD
1.0000
0.0000
0.5000
0.5000
0.8333
0.1667
0.5000
0.5000
1.0000
0.0000
0.5714
0.4286
0.5000
0.5000
0.6000
0.4000
1.0000
0.0000
1.0000
0.0000
0.8000
0.2000
1.0000
0.0000

Figure 3.4

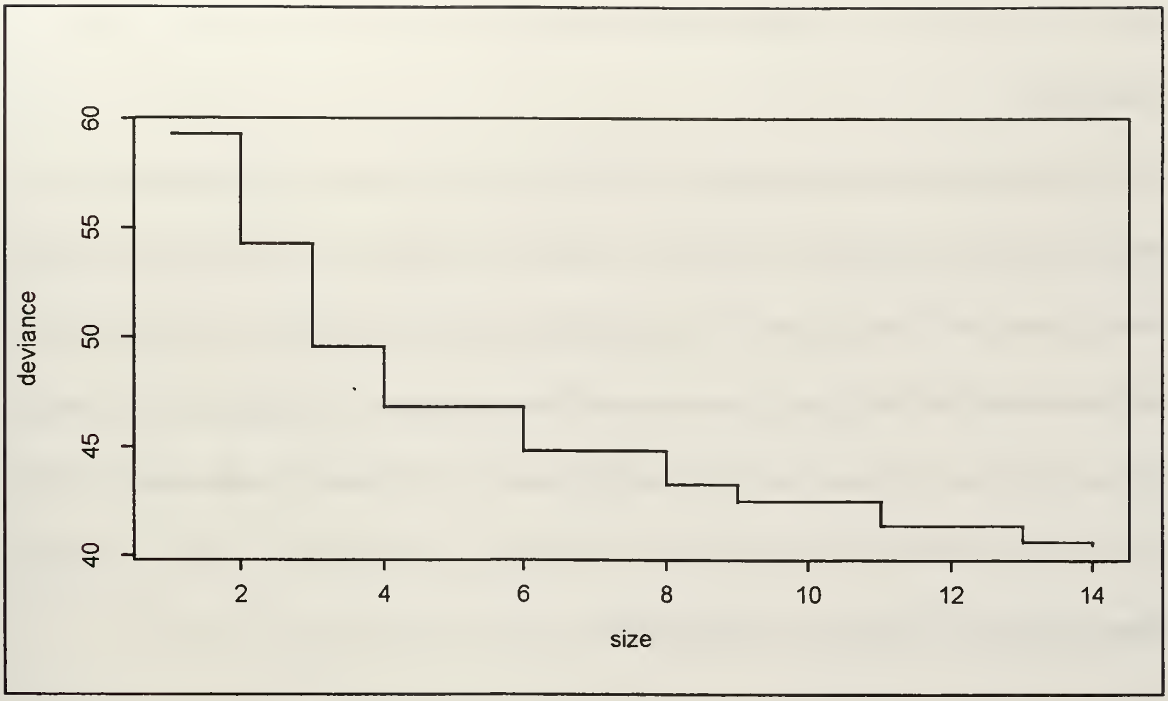


Figure 3.5 Pruning Plot for Example

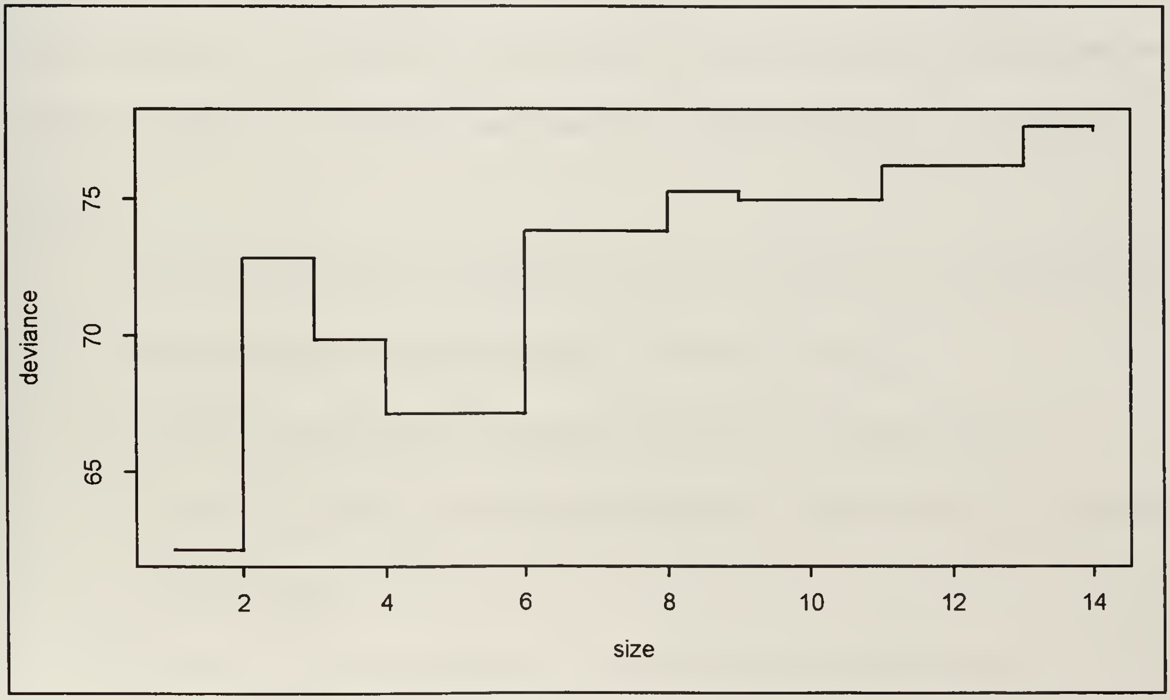


Figure 3.6 Cross-Validation Plot for Example

chosen. An additional consideration in selecting this tree size is its ease and clarity of presentation.

By executing the pruning method with inputs that include the overgrown tree and a tree size of four, a tree that is similar to that found in Figure 3.3 can be plotted. The reader is reminded that Figure 3.3 contains information that is not normally found on a plot of a pruned tree. Figure 3.3 was created by executing the pruning method in S-Plus with a tree size of four and then adding some supplementary information. The supplementary information was found by executing some additional S-Plus commands. These commands are included in Appendix D.

IV. ANALYSIS AND RESULTS

The data file is made up of 32,978 soldiers. One major difference between this data file and the data file used in the example is that the Loss category has four levels instead of two. As described in Chapter II, the four Loss levels are: EAdv (out early for adverse reason), EOK (out early for reason other than adverse), EndT (out at the end of the first-term of enlistment), and Not (stayed in past the end of the first-term of enlistment).

Analysis will be performed on two different formats of the data file. One format of the data file will be referred to as the "C*-Group" data and the other format will be referred to as the "Regular" data. The only difference between the two formats is the way the attributes are partitioned. Analysis will be conducted on both formats, beginning with the C*-Group data. Techniques introduced in the CART example will be used to perform the analysis. The CART example reveals specific steps for conducting the analysis. These steps are:

- Chose the attributes (independent variables) that will be used to construct tree.
- Build an overgrown tree to reveal the structure of the data.
- Create the pruning and cross-validation plots for the overgrown tree.
- Review the pruning and cross-validation plots. Use the plots to select the "best" tree size.
- Grow a tree that is pruned to the "best" size selected in previous step.
- Review the results.

A. C*-GROUP DATA

The C*-Group data differs in its format from the Regular data only by the number of levels each attribute contains. Five attributes make up the C*-Group data. Those attributes are AFQT, Gender, EdGrp, Term, and Race. In the C*-Group data, the attribute AFQT consists of only three levels: I-IIIA, IIIB, and IV-V. The attributes Gender, EdGrp, and Term each have only two levels. Gender consists of male and female, EdGrp consists of HSD and NoHSD, and Term consists of 3&4Yrs and Other. The attribute Race consists of three levels: White, Black, and Other. If Race is omitted, the format of the data closely resembles the attribute levels used in constructing the current C-Groups. The first 40 rows of the C*-Group data and the important S-Plus commands used during this portion of the analysis can be found in Appendix E.

1. C*-Group Data With Four Attributes

Only four attributes are used to begin the analysis of the C*-Group data. The attributes AFQT, Gender, EdGrp, and Term are selected because they match the attributes used in the construction of the current C-Groups. These four attributes are then used to create a overgrown tree. Due to the size of the data file, the stopping criteria will be left in place. A tree created from the C*-Group data is displayed in Figure 4.1. Although it may be difficult to see, this tree has 16 terminal nodes. This will be considered to be an overgrown tree. Since this is not the final tree for this portion of the analysis, information was purposely omitted from Figure 4.1. The next step in the process is to look at the pruning and cross-validation plots.

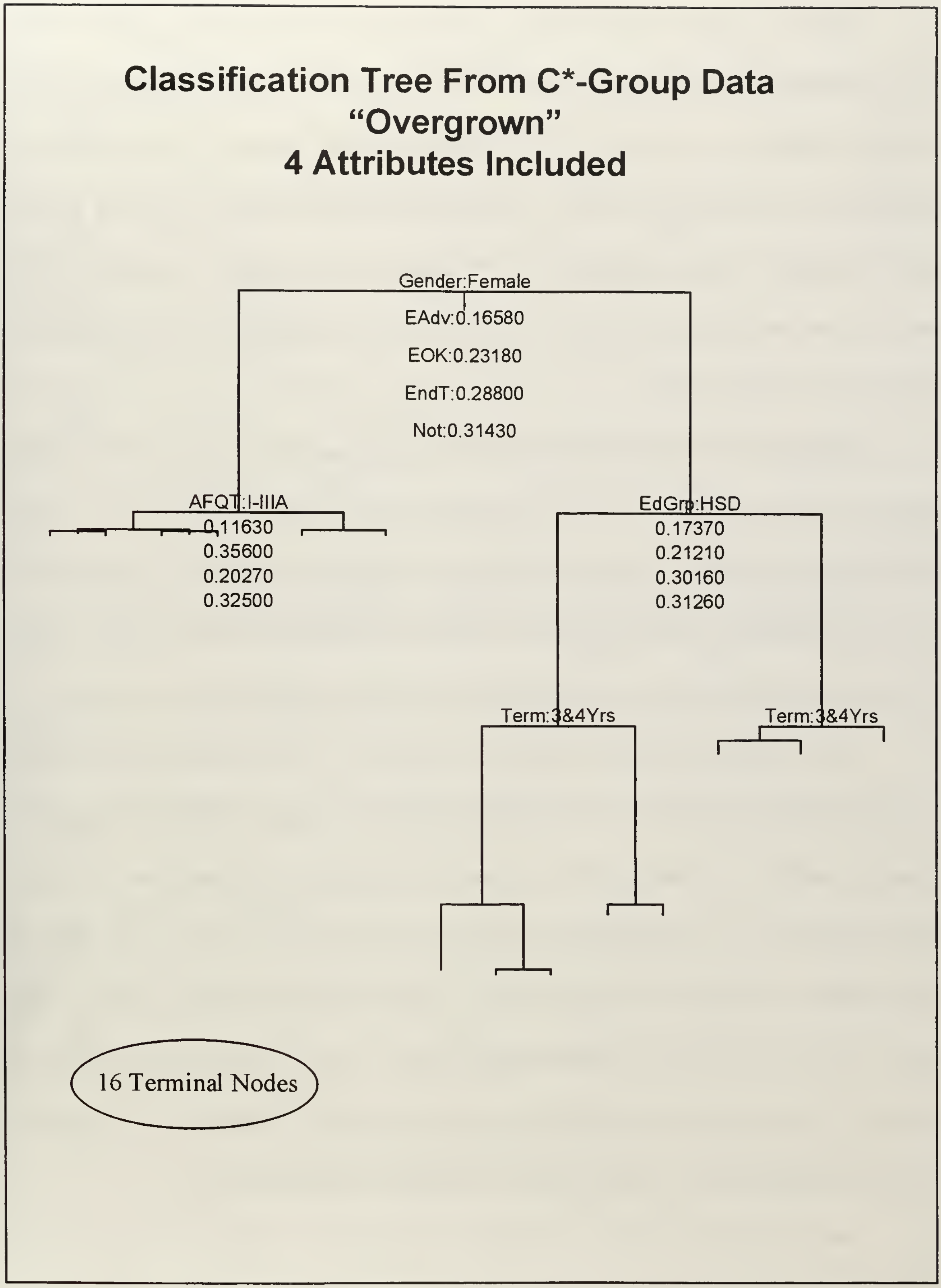


Figure 4.1 Overgrown Classification Tree Using C*-Group Data With 4 Attributes

Figure 4.2 is the result of executing the pruning method with the overgrown tree as the only input. No tree size was provided to the method. The plot indicates that the deviance continues to decrease as the tree size increases. Figure 4.3 is the result of executing the cross-validation method. This plot indicates that after a tree size of 10, the deviance begins to increase, although at a very slow rate. By looking at the pruning plot and the cross-validation plot together, one can see that a tree size of 10 terminal nodes would be an excellent choice.

Figure 4.4 is a tree created from the C*-Group data that has been pruned back to the best 10 terminal nodes. Each node's number has been placed inside a diamond. The diamonds for the terminal nodes have been placed underneath the node. The proportion of the total number of cases found in each Loss level is printed below each node. Lastly, printed below the proportions of Loss levels is the number of cases found at each node.

There is a great amount of information in Figure 4.4. The splitting criterion at the root node is based on Gender. This indicates that by executing CART in S-Plus, the greatest reduction in deviance will be achieved by splitting on Gender first. In other words, Gender is the most significant attribute contributing to the purity of the terminal nodes. After receiving all the females, node 2 is then split on the attribute AFQT. The split at node 2 produces two terminal nodes, nodes 4 and 5. The classification tree in Figure 4.4 is indicating that AFQT is the only attribute that determines the Loss type proportions for females. Length of term and education do not play a role.

Terminal node 24 in Figure 4.4 contains 12,192 cases. At 37% (12,192 / 32,978), this node accounts for the largest number of cases in a terminal node. The terminal node

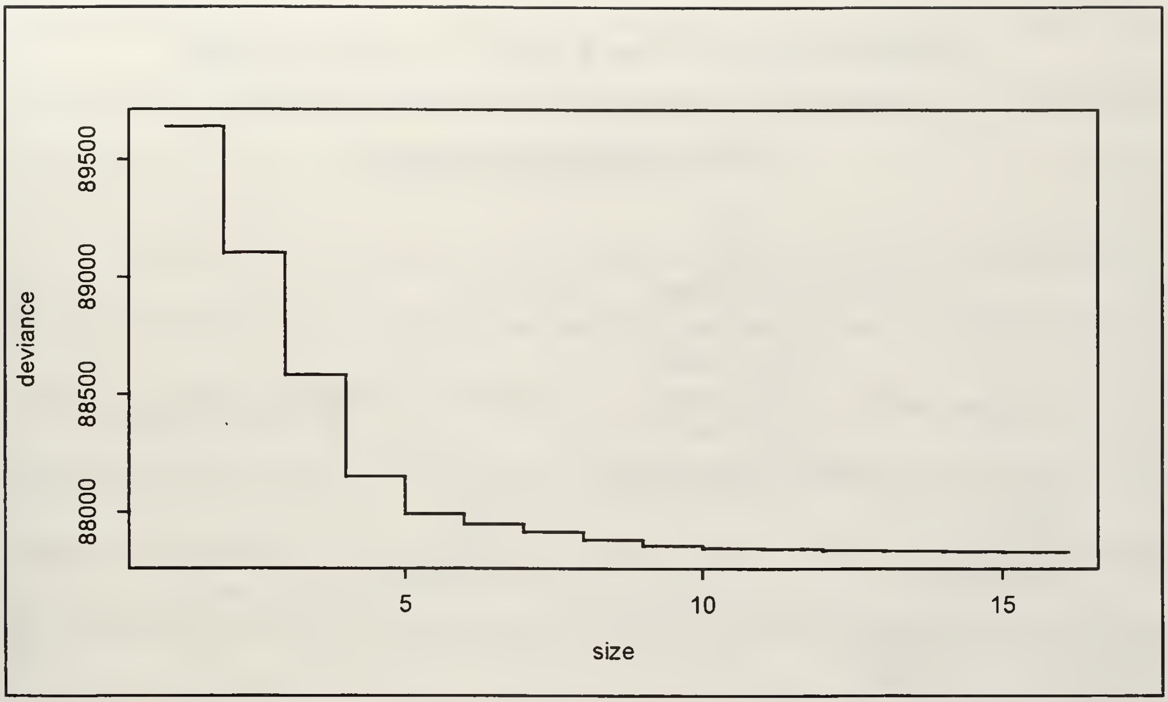


Figure 4.2 Pruning Plot When 4 Attriburtes Included Using C*-Group Data

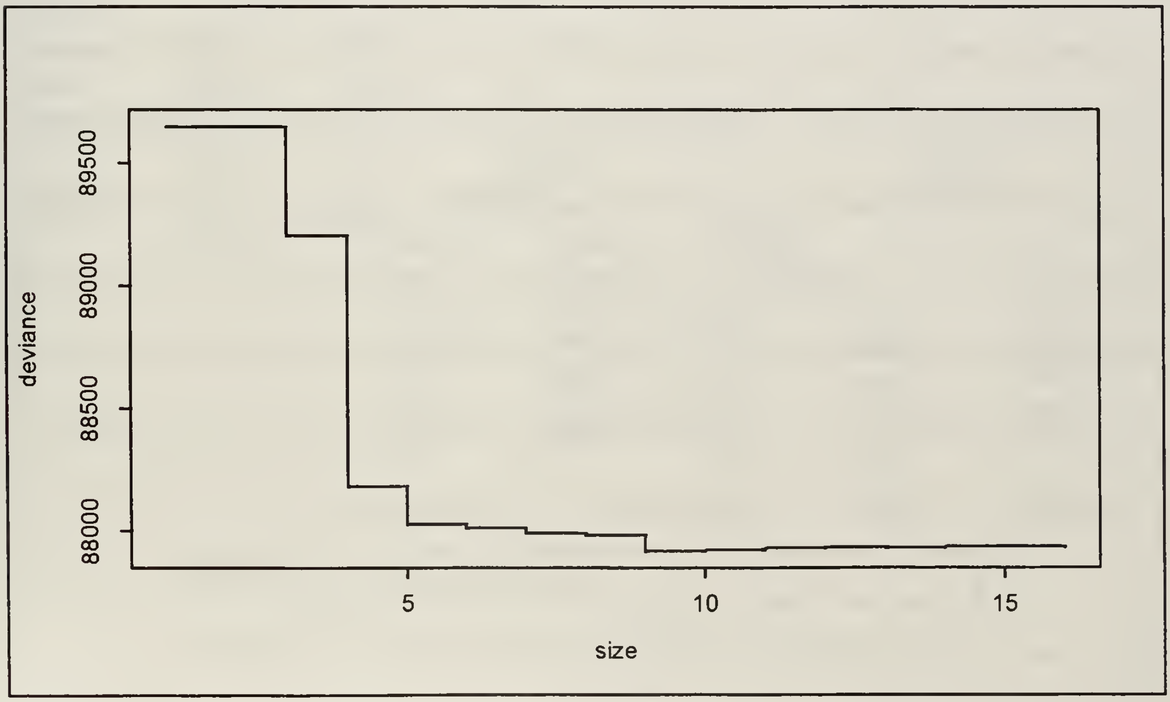


Figure 4.3 Cross-Validation Plot When 4 Attributes Included Using C*-Group Data

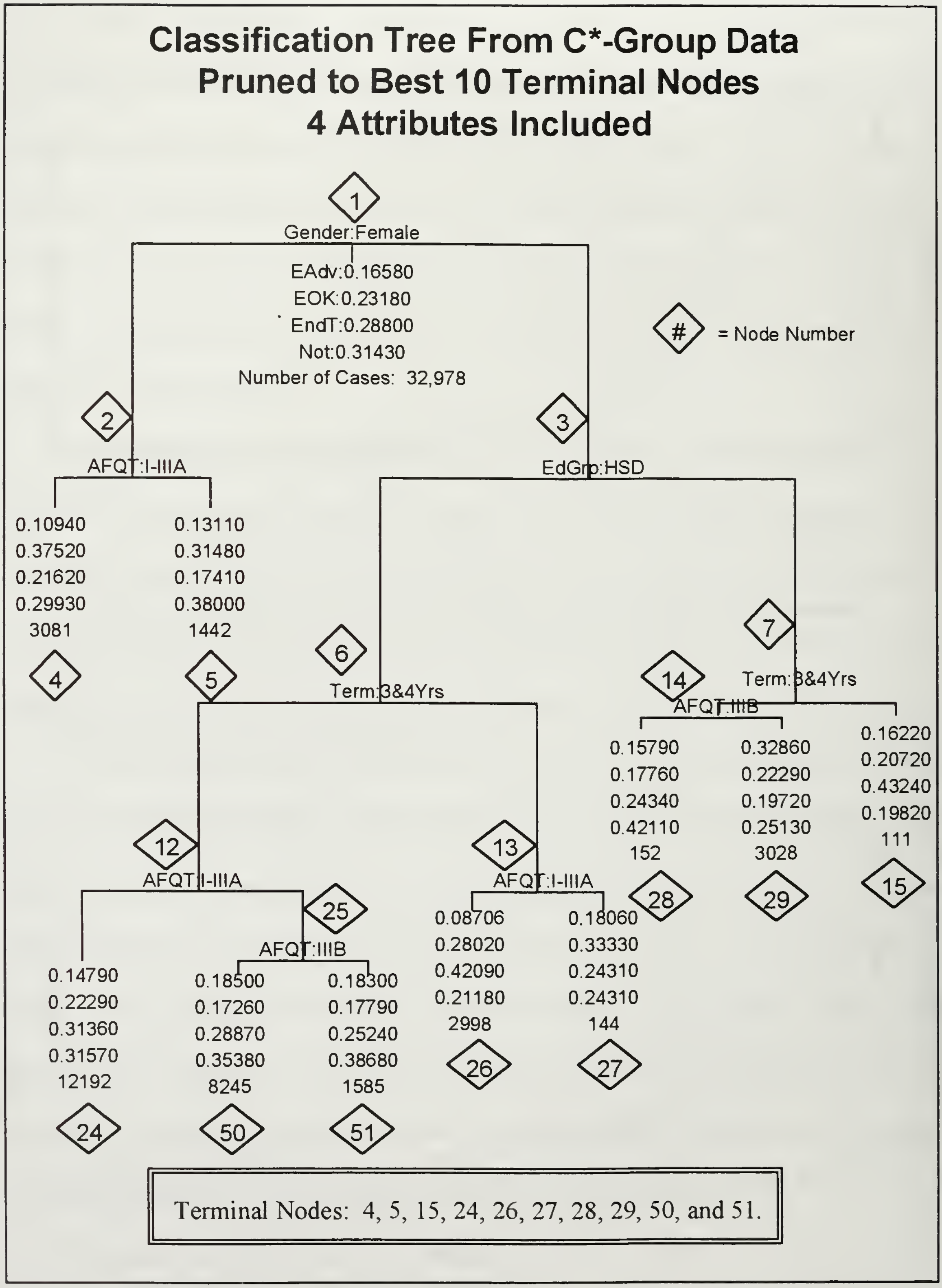


Figure 4.4 Pruned Classification Tree Using C*-Group Data With 4 Attributes

with the next largest number of cases is node 50 and this node accounts for only 25% of the total number of cases. Since such a large proportion of the total number of cases falls into node 24, the node should be examined further.

Beginning at the root node and following the splits in the tree, the attribute levels that characterize the cases in node 24 are: males; high school degree or better; enlisted for term of 3 or 4 years; and belong in AFQT category I, II, or IIIA. These attributes exactly match the attributes of the Army C-Group 1 (see Figure 1.1). The presence of four Loss categories allows the exploration of combinations of the categories. By breaking down the Loss category into four types, one can see that the proportion of cases that fall in the Loss type of Not is 0.3157. That is, approximately 32% of the soldiers in node 24 stay in past the end of their first term of enlistment. Also, approximately 31% of the soldiers complete their first term of enlistment and then separate from the Army. Combining these two figures, approximately 63% of the soldiers in node 24 meet or exceed the length of their enlistment contracts. This information is not available when there are only two Loss types present. When there are only two Loss types present, the first three Loss types in Figure 4.4 are summed together to form a single type called “lost.” As a result, there would only be two Loss types, called “lost” and “not.” The proportion of cases from node 24 that are in type “lost” is 0.6843, or approximately 68%. The proportion that are in type “not” is 0.3157, or approximately 32%. Information about the soldiers who do not stay in past their first term of enlistment is lost!

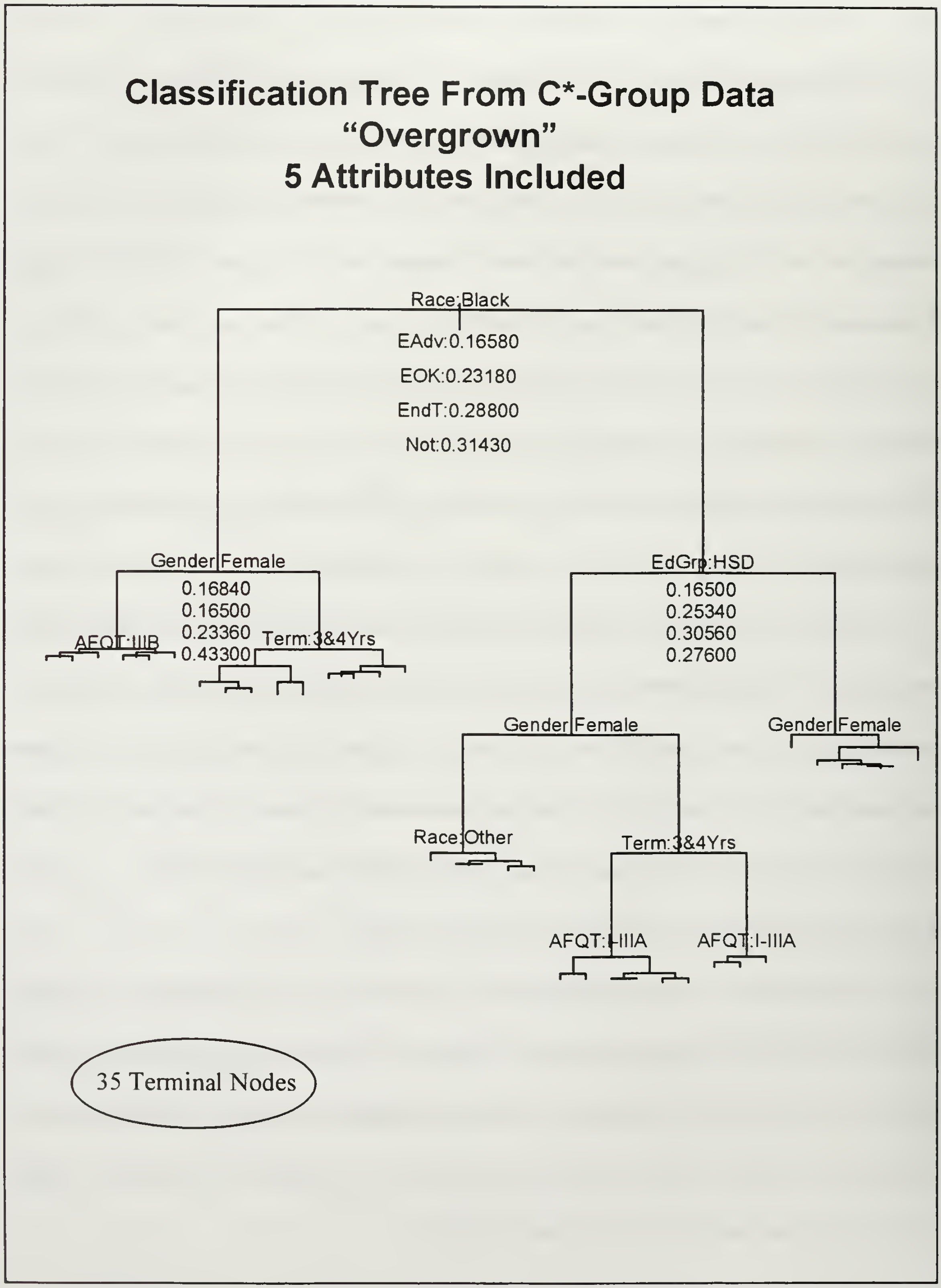
The cost of the additional information with four Loss types is an increase in the misclassification rate. With four Loss types present, node 24 is classified as Not (the level with the largest proportion of cases). The misclassification rate of node 24, 0.6843, is the sum of the proportions assigned to the remaining levels. With two Loss types present, node 24 would be classified as "lost" and the misclassification rate would be 0.3157. One can easily see that the misclassification rate is lower when only two Loss types are used. However, the high misclassification rate achieved when four Loss types are used can be reduced by grouping the levels appropriately. Suppose the Army wants know what soldiers are leaving early, that is, are separating from the service prior to completing their first term of service. Using the four Loss types, one would want to group EAdv and EOK into a single type called "early." EndT and Not would be grouped into a single type called "notEarly." As a result, node 24 would be classified "notEarly" and the misclassification rate would be 0.3708. This is a significant reduction from the earlier value of 0.6843.

One must remember that the number of cases in node 24 is larger than any other terminal node. This is an important fact when examining the Loss type proportions at the node. Looking at the proportions without looking at the number of cases in node can be deceiving. For example, only 15% of the soldiers in node 24 are lost for an adverse reason. This may seem like a very favorable figure. In fact, only three other terminal nodes have a lower percentage for EAdv. However, the 15% in node 24 accounts for 1,803 soldiers. Of all the terminal nodes, node 24 has the largest number of soldiers lost for adverse reasons. Other nodes can be examined in a similar fashion.

2. C*-Group Data With Five Attributes

The attribute Race is now introduced to the analysis. How large an impact will this attribute have on the analysis? The same analysis steps used to arrive at Figure 4.4 are now repeated. Race is included in the list of attributes used to create Figure 4.5, the initial overgrown tree. This overgrown tree has 35 terminal nodes. Figures 4.6 and 4.7 are the Pruning and Cross-Validation Plots, respectively. The "best" tree size is not as clear for five attributes as it was with four attributes. One could easily argue that a tree size of 18 might be appropriate. However, for ease of presentation and comparison to the tree grown with four attributes, a tree size of 9 is chosen. This size tree was chosen because a tree with 10 terminal nodes, in this case, does not result in reducing the deviance. In fact, the deviance for a tree with 10 terminal nodes is the same as the deviance for a tree with 9 terminal nodes. When deviance remains the same, the smaller tree size is selected. A tree pruned to the 9 "best" terminal nodes is found in Figure 4.8. Examination of this tree indicates that if any one of the 9 terminal nodes is split, there will be two additional terminal nodes. A tree with the 10 "best" terminal nodes can not be created. The same presentation methods used in Figure 4.4 are used in Figure 4.8.

The impact of including the attribute Race in the analysis is dramatic. The largest reduction in deviance initially achievable is realized by splitting on the attribute Race. The attributes that contribute the most to reducing the deviance of a tree are the ones to include in the analysis. If the number of attributes must be limited to less than five, Figure 4.8 clearly shows that Race must be one of the five.



Classification Tree From C*-Group Data
"Overgrown"
5 Attributes Included

35 Terminal Nodes

Figure 4.5 Overgrown Classification Tree Using C*-Group Data With 5 Attributes

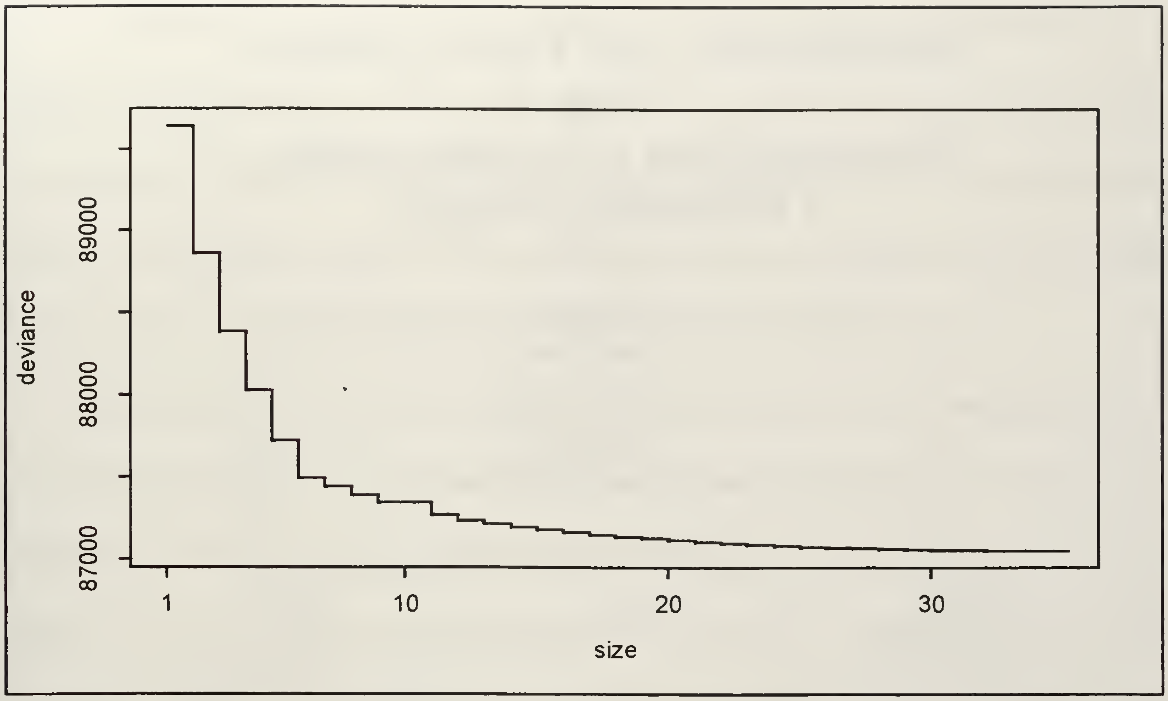


Figure 4.6 Pruning Plot When 5 Attriburtes Included Using C*-Group Data

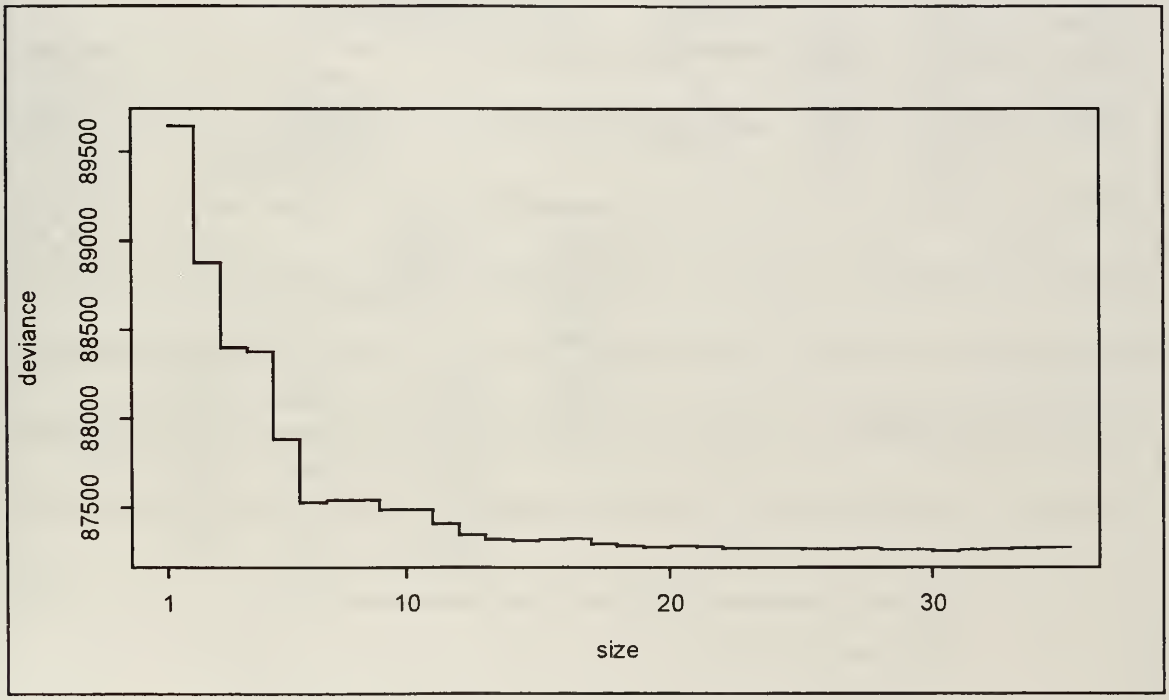


Figure 4.7 Cross-Validation Plot When 5 Attributes Included Using C*-Group Data

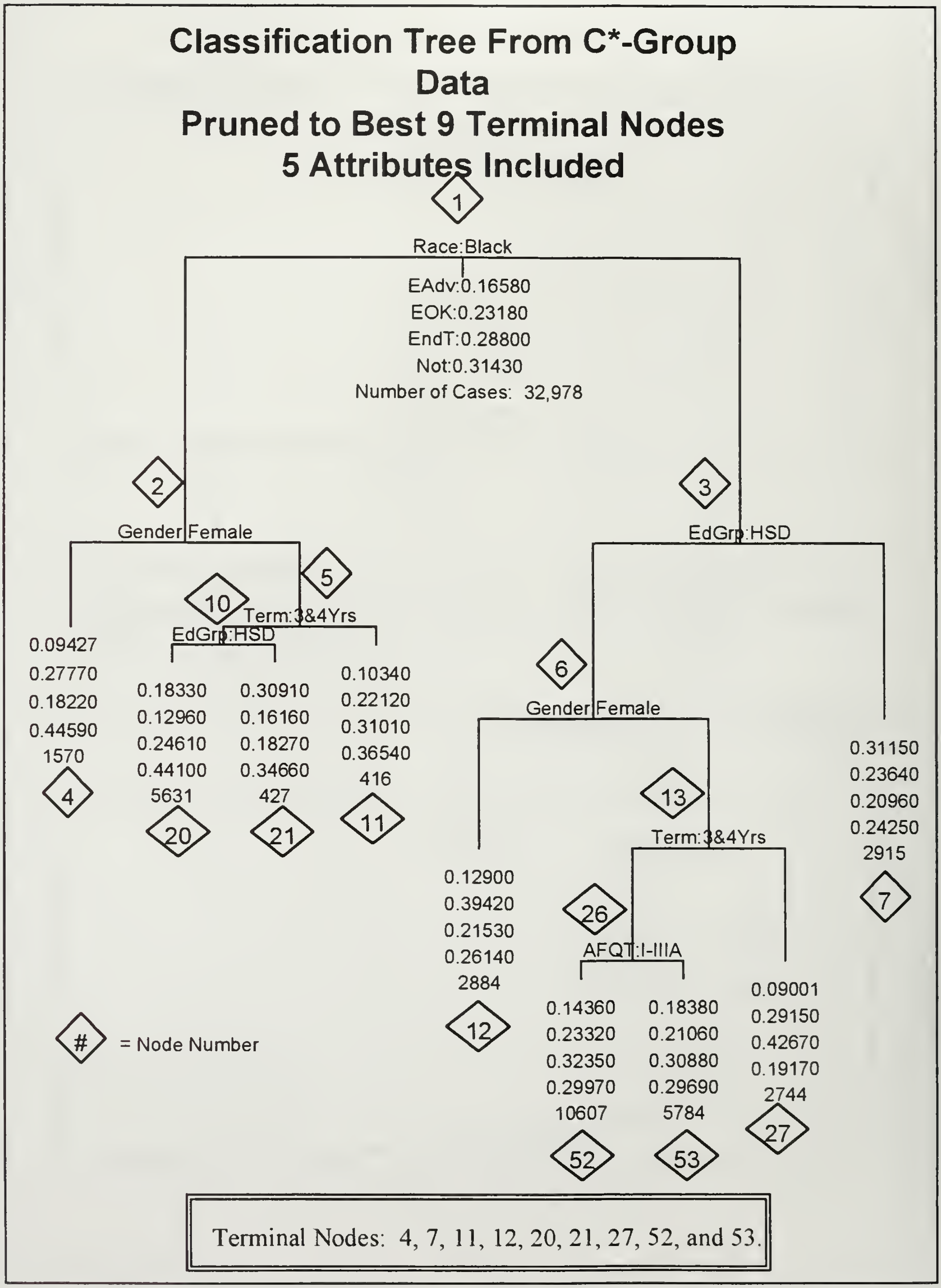


Figure 4.8 Pruned Classification Tree Using C*-Group Data With 5 Attributes

In Figure 4.8, the terminal node containing the largest number of cases is node 52. Node 52 has all the attribute characteristics of C-Group 1 with one exception. The node only contains the Race levels of White and Other. This node accounts for approximately 32% of all total number of cases. The terminal node with the next largest number of cases is node 53 but it only accounts for 18% of the total number of cases.

Exploring the available information on the tree in Figure 4.8 in a similar method as was used with Figure 4.4, one can again see the benefit of having four Loss type categories. With only two Loss type categories, 70% of the cases in node 52 would be placed in the "lost" category and 30% in the "not" category. Having four Loss type categories available indicates that 62% of the cases in node 52 meet or exceed the contracted term length. Only 38% detach the Army before their term has expired.

Again, the size of node 52, in terms of the number of cases it contains, is very important. Of the 38% who leave the Army early, 23% leave for non-adverse type reasons. The Army usually has very little control over the soldiers who detach early for other than adverse reasons. The soldiers who detach for adverse reasons accounts for only 14% of the number of cases in node 52. However, the 14% includes 1,523 soldiers that detach for adverse reasons. Of the 9 terminal nodes, node 52 contains the largest number of soldiers detaching for adverse reasons. The figure of 14% may seem low, but when combined with the number of cases in the node, the result produces a significant figure.

B. REGULAR DATA

Like the C*-Group data, five attributes are found in the Regular data. The difference between the two data files is the number of levels present in the attributes. The attributes present are AFQT, Gender, EdGrp, Term, and Race. In the Regular data, the attribute AFQT has six levels: I, II, IIIA, IIIB, IV, and V. The attribute Gender still consists of just two levels, male and female. The EdGrp attribute has five levels in the Regular data. The five EdGrp levels are NoHSD, GED, HSD, <=2YrsColl, and >2YrsColl. The five levels of the attribute Term are 2Yrs, 3Yrs, 4Yrs, 5Yrs, and 6Yrs. The three levels of the attribute Race are the same as in the C*-Group data, that is, Black, White, and Other. The first 40 rows of the Regular data are identical to the data in Appendix B. The important S-Plus commands used with the Regular data can be found in Appendix F.

1. Regular Data With Four Attributes

Analysis of the Regular data follows the same steps used during the analysis of the C*-Group data. Initially, only four attributes will be used. The four attributes used are AFQT, Gender, EdGrp, and Term. An overgrown tree is created. This tree is displayed in Figure 4.9. The overgrown tree has 68 terminal nodes. Pruning and cross-validation plots are now constructed. These plots are presented in Figures 4.10 and 4.11, respectively. The pruning plot shows a steady decline in deviance as the tree size increases. The deviance in the cross-validation plot is constantly decreasing until it reaches a minimum at a tree size of 38 terminal nodes. Since a tree of this size is probably too large to be useful, a smaller tree size must be selected. Although the deviance is

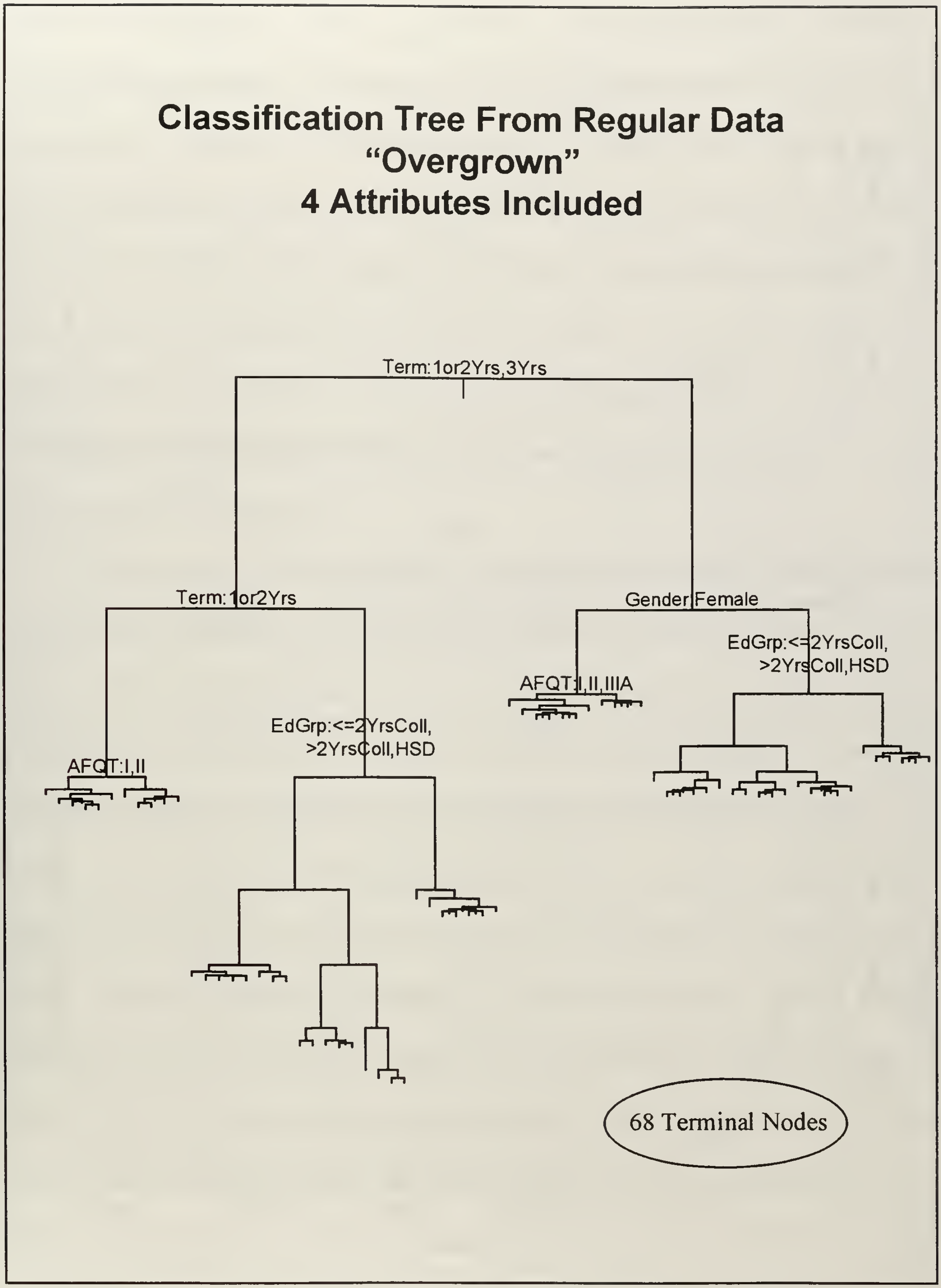


Figure 4.9 Overgrown Classification Tree Using Regular Data With 4 Attributes

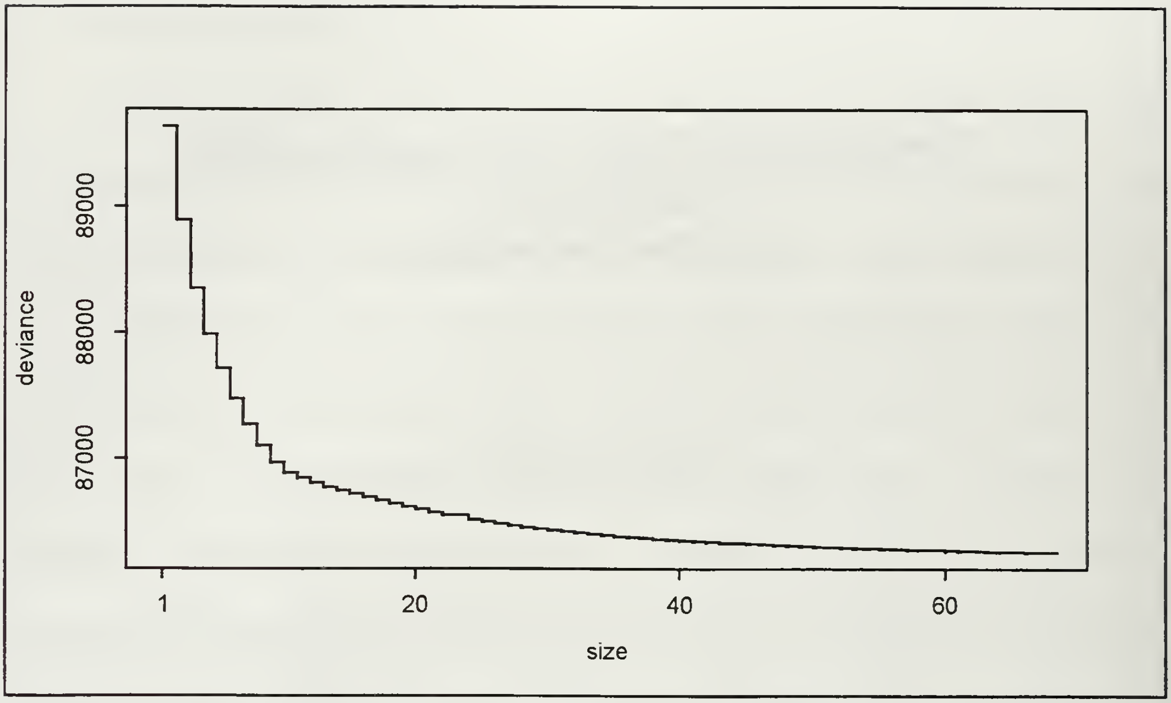


Figure 4.10 Pruning Plot Using Regular Data When 4 Attriburtes Included

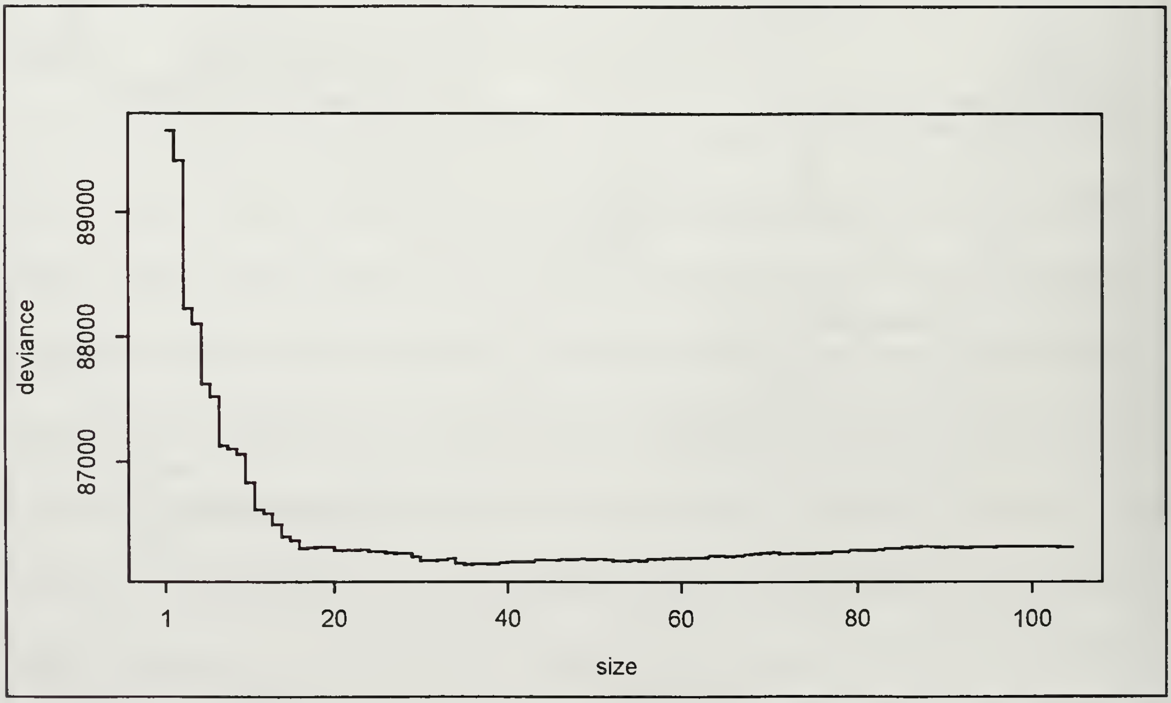


Figure 4.11 Cross-Validation Plot Using Regular Data When 4 Attributes Included

constantly decreasing up to a size 38, a good place to look is where the rate of decrease in deviance begins to decline. In this case, the place to look is around a tree size of 15 terminal nodes. A tree size of 15 terminal nodes could be selected as the optimal size. In previous analysis, the optimal tree size selected was at or near 10 terminal nodes. Since a tree size of 15 terminal nodes is larger than the pruned trees previously presented, two pruned trees will be presented here. Displayed in Figure 4.12 is a tree that has been pruned to the best 15 terminal nodes. Due to space limitations, only the root node and terminal nodes have been numbered. A tree that has been pruned to the best 10 terminal nodes is displayed in Figure 4.13.

Which tree size is more appropriate? Arguments can be made for both tree sizes. The larger tree size breaks down the number of cases into additional categories. This can lead to a finer detail of information. On the other hand, the size of the terminal nodes, as measured by the number of cases contained in the node, may be too small. For example, node 22 in Figure 4.12 contains only 63 cases, or approximately 0.2% of the total number of cases. A terminal node of this size may indicate that the tree size is too big. In fact, Figure 4.12 contains six terminal nodes that each contain 3% or less of the total number of cases. How much additional information is obtained from having a greater number terminal nodes if those nodes only contain a very small percentage of the total number of cases? The answer to this question will vary depending on the situation and the goals that were established at the beginning of the process. Figure 4.13 presents a smaller tree, one with 10 terminal nodes. Node 86 in this smaller tree contains only 0.8% of the total number of cases. Although this node is only slightly larger than the smallest node in

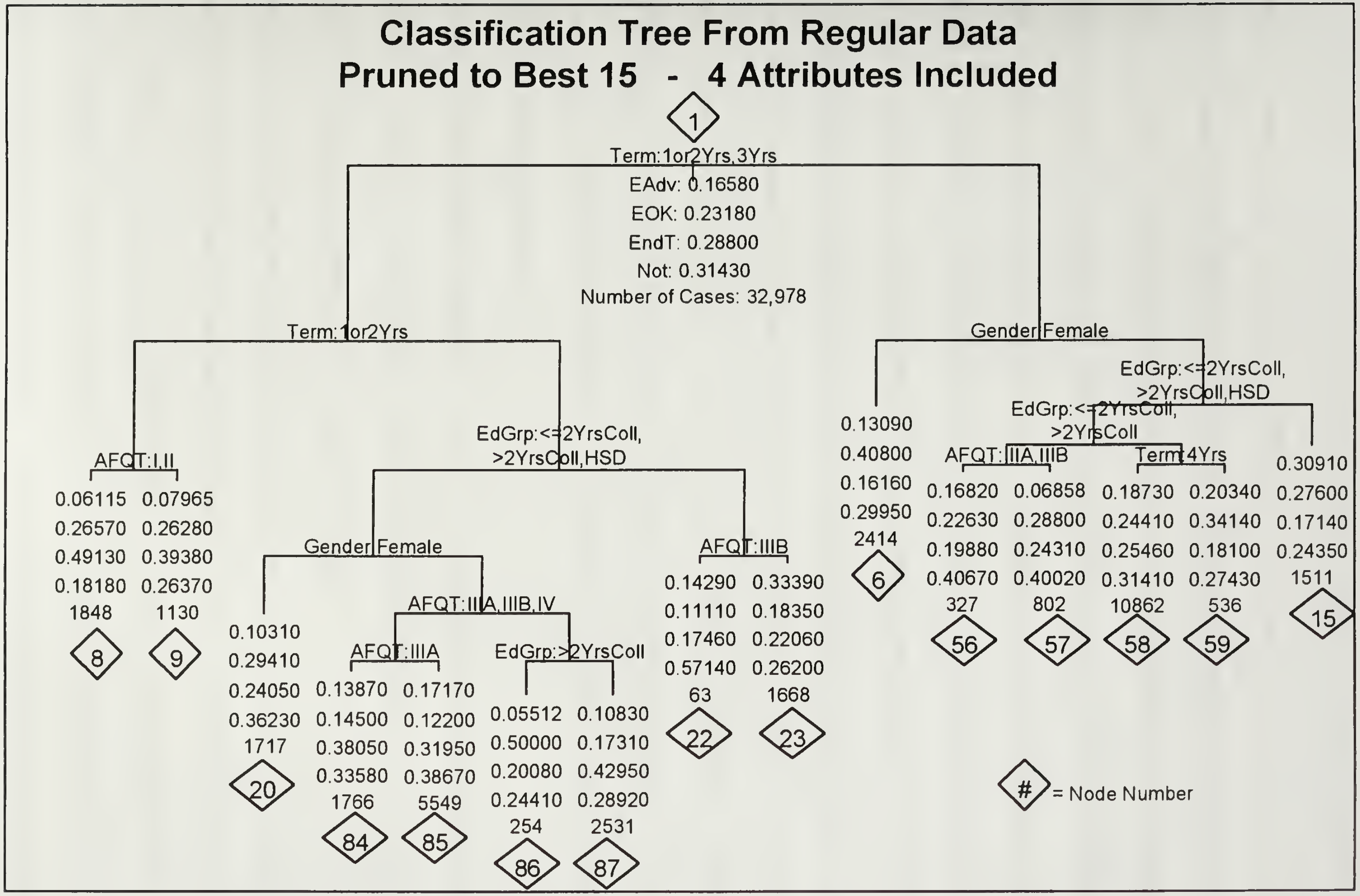


Figure 4.12 Pruned Classification Tree Using Regular Data With 4 Attributes

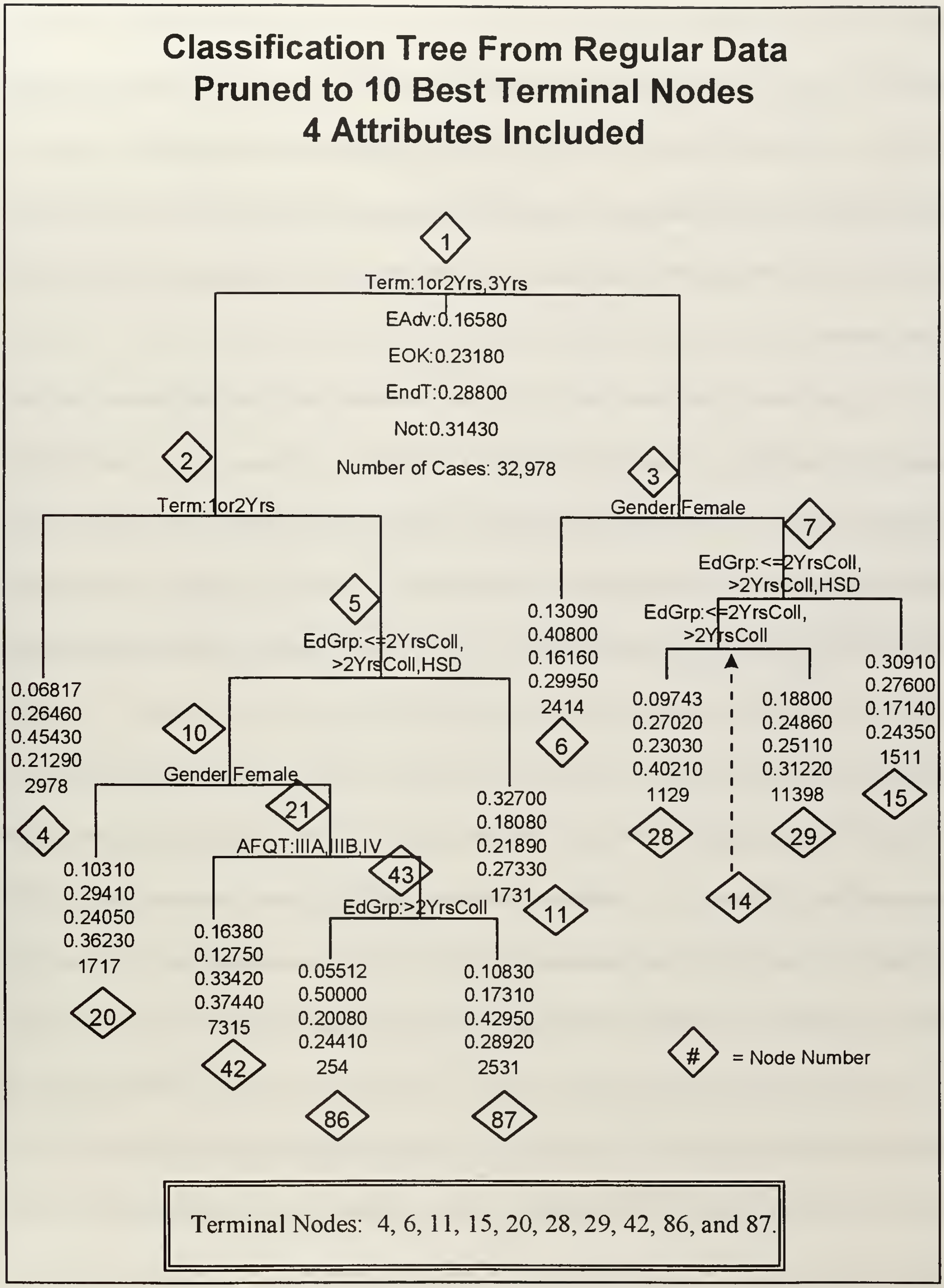


Figure 4.13 Pruned Classification Tree Using Regular Data With 4 Attributes

Figure 4.12, there are only two nodes in Figure 4.13 that contain 3% or less of the total number of cases. An initial evaluation is that a tree size of 10 terminal nodes is sufficient in this case. Selecting a smaller tree size is not always the correct choice. Each process is unique. Each process has its own concerns and measures of effectiveness. The availability of even a small amount additional information may have a dramatic impact under certain circumstances.

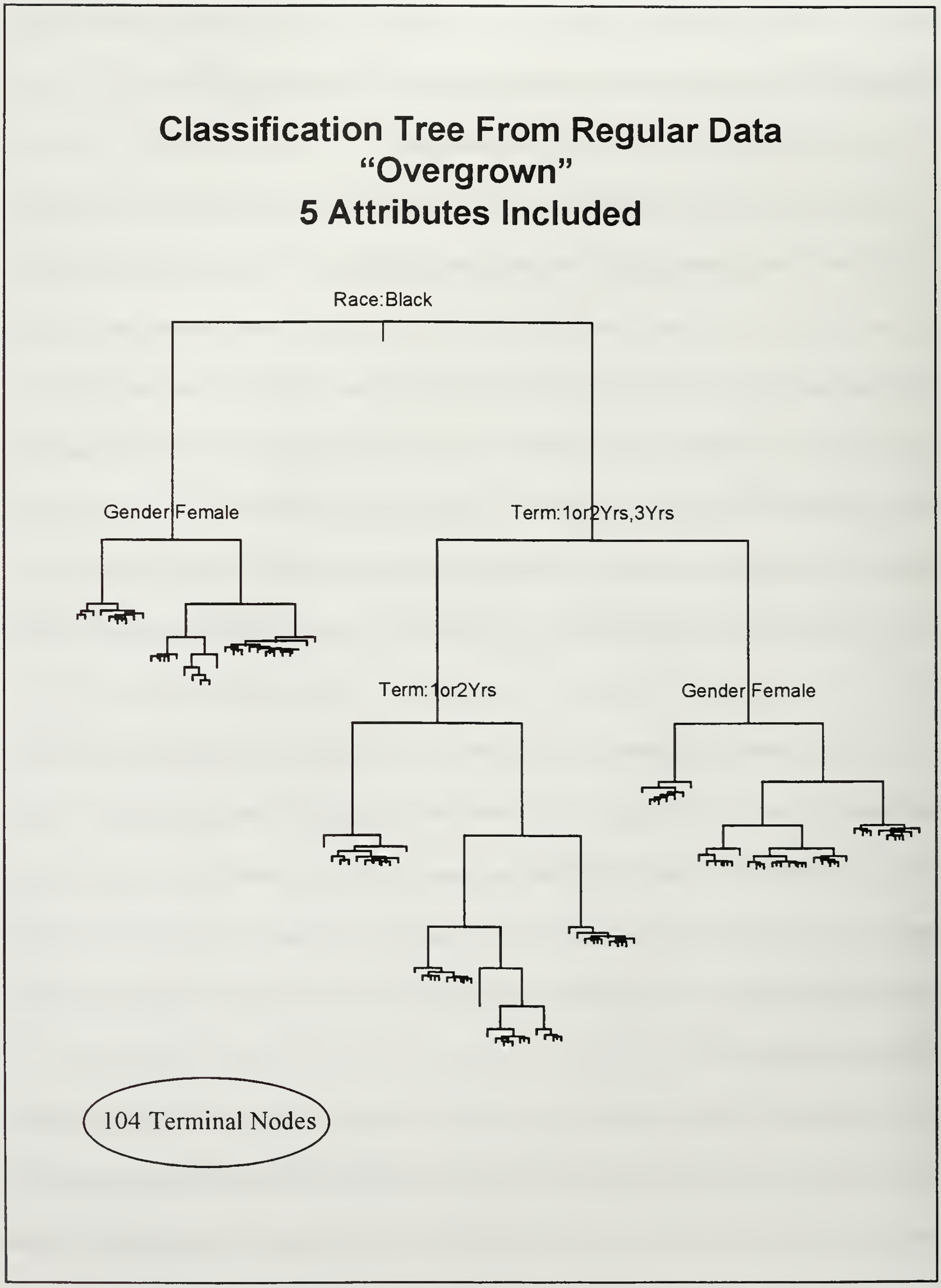
Regardless of what tree size is selected, examining the terminal nodes provides insights into the data. Terminal nodes that contain very few cases have already been discussed. Similar to the analysis performed on the C*-Group data, terminal nodes that contain large number of cases should be investigated. Node 29 in Figure 4.13 contains 11,398 cases, or 35% of the total number of cases. The cases in node 29 are made up of males with a high school degree that have enlisted for a term of 4, 5, or 6 years. The path of nodes from the root node to node 29 is: 1 to 3 to 7 to 14 to 29. At node 7 the split is made on EdGrp. At node 14 the split is also made on EdGrp. This is a key point in the analysis of trees. Node 14 can only split the levels of EdGrp it has received from node 7. Specifically, at node 7 all cases with a high school degree or more are placed in the left child node, node 14. At node 14 all cases with any college education are placed in node 28. All other levels of EdGrp present at node 14 are placed in node 29. The only levels of EdGrp that came into node 14 were HSD, <=2Yrscoll, and >2YrsColl. The only EdGrp level being placed in node 29 is HSD. Multiple splits on the same attribute can be accomplished, in part, because EdGrp in the Regular data contains five levels. EdGrp in the C*-Group data contained only two levels. Having five levels of EdGrp present in the Regular data results in providing addditional information in the tree structure.

2. Regular Data With Five Attributes

The analysis steps will now be executed using the Regular data and five attributes, i.e., Race will be included. The overgrown tree in Figure 4.14 has 104 terminal nodes. Figures 4.15 and 4.16 present the pruning and cross-validation plots, respectively. These plots are very similar to those observed when only four attributes were used with the Regular data. Following the established reasoning presented, two pruned trees are created. Figure 4.17 displays a tree that has been pruned to the "best" 15 terminal nodes and Figure 4.18 displays a tree that has been pruned to the "best" 10 terminal nodes. The presentation methods used for Figures 4.12 and 4.13 are also used for Figures 4.17 and 4.18.

When the attribute Race was added to the C*-Group data analysis, it became the attribute split on at the root node. Since the levels in the attribute Race are the same in both data formats, one would expect an outcome of including Race in the Regular data analysis to be similar to the result of the C*-Group data analysis. This happens and at the root node. Of the five attributes used, splitting on Race contributes the most to reducing the deviance of the tree.

Should the final tree size be 10 or 15 terminal nodes? The earlier discussion of this topic holds here also. The final tree size will be determined by the situation, the process, the goals established, and the measures of effectiveness selected. The remainder of this analysis will concentrate on the smaller tree in Figure 4.18.



Classification Tree From Regular Data
"Overgrown"
5 Attributes Included

104 Terminal Nodes

Figure 4.14 Overgrown Classification Tree Using Regular Data With 5 Attributes

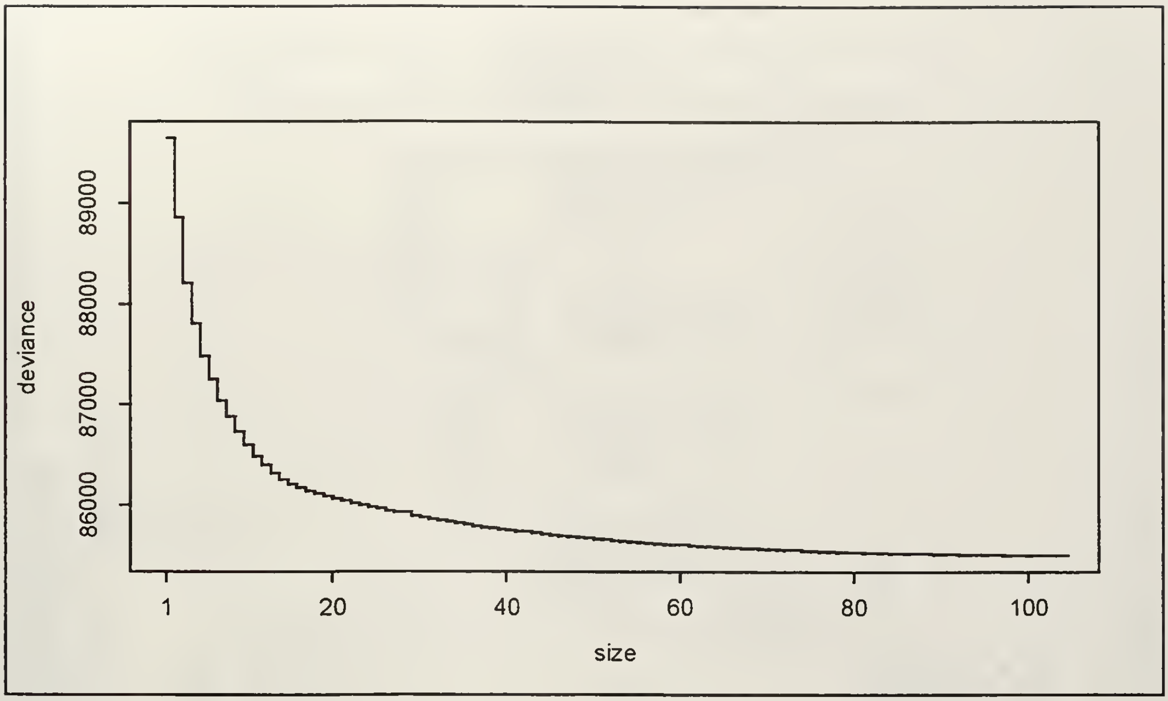


Figure 4.15 Pruning Plot Using Regular Data When 5 Attributes Included

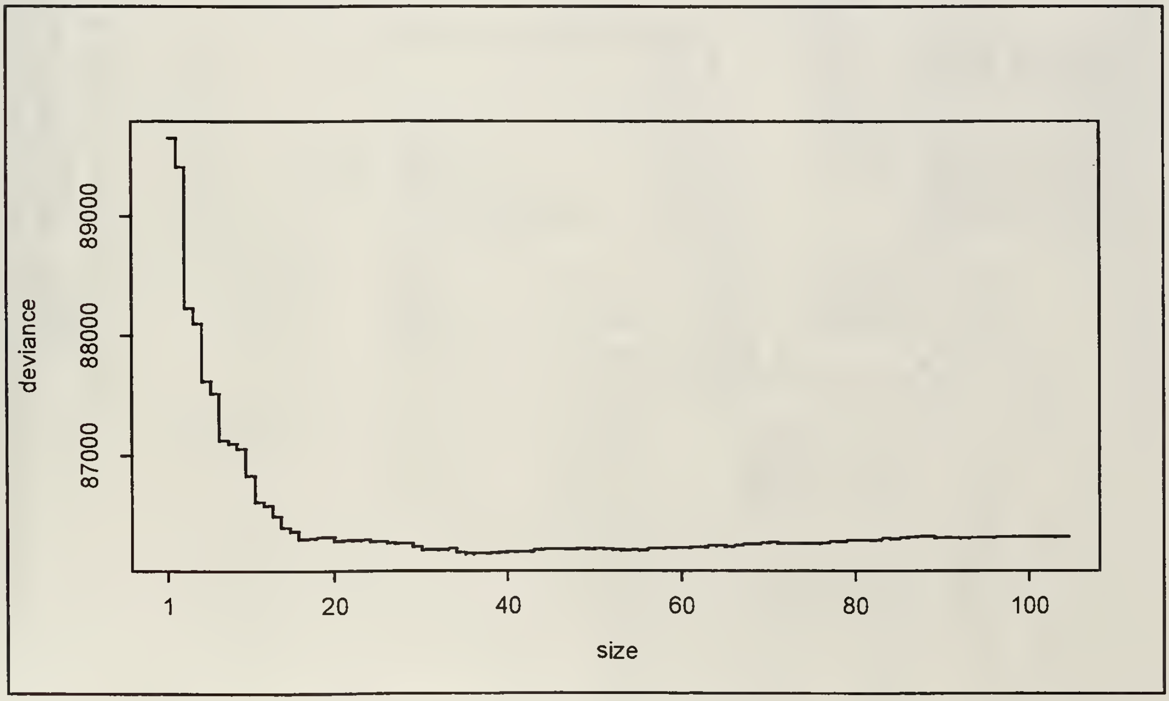


Figure 4.16 Cross-Validation Plot Using Regular Data When 5 Attributes Included

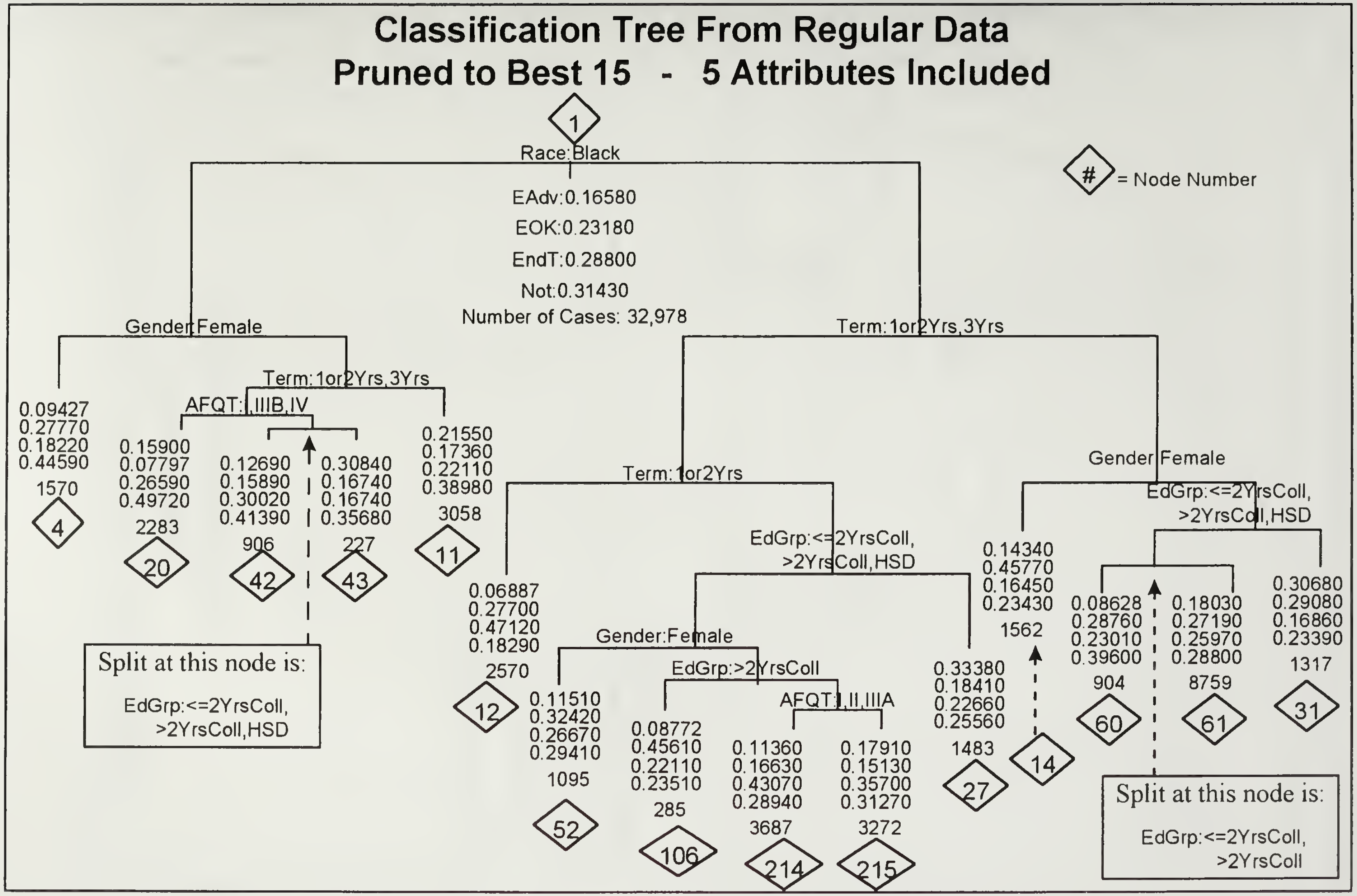


Figure 4.17 Pruned Classification Tree Using Regular Data With 5 Attributes

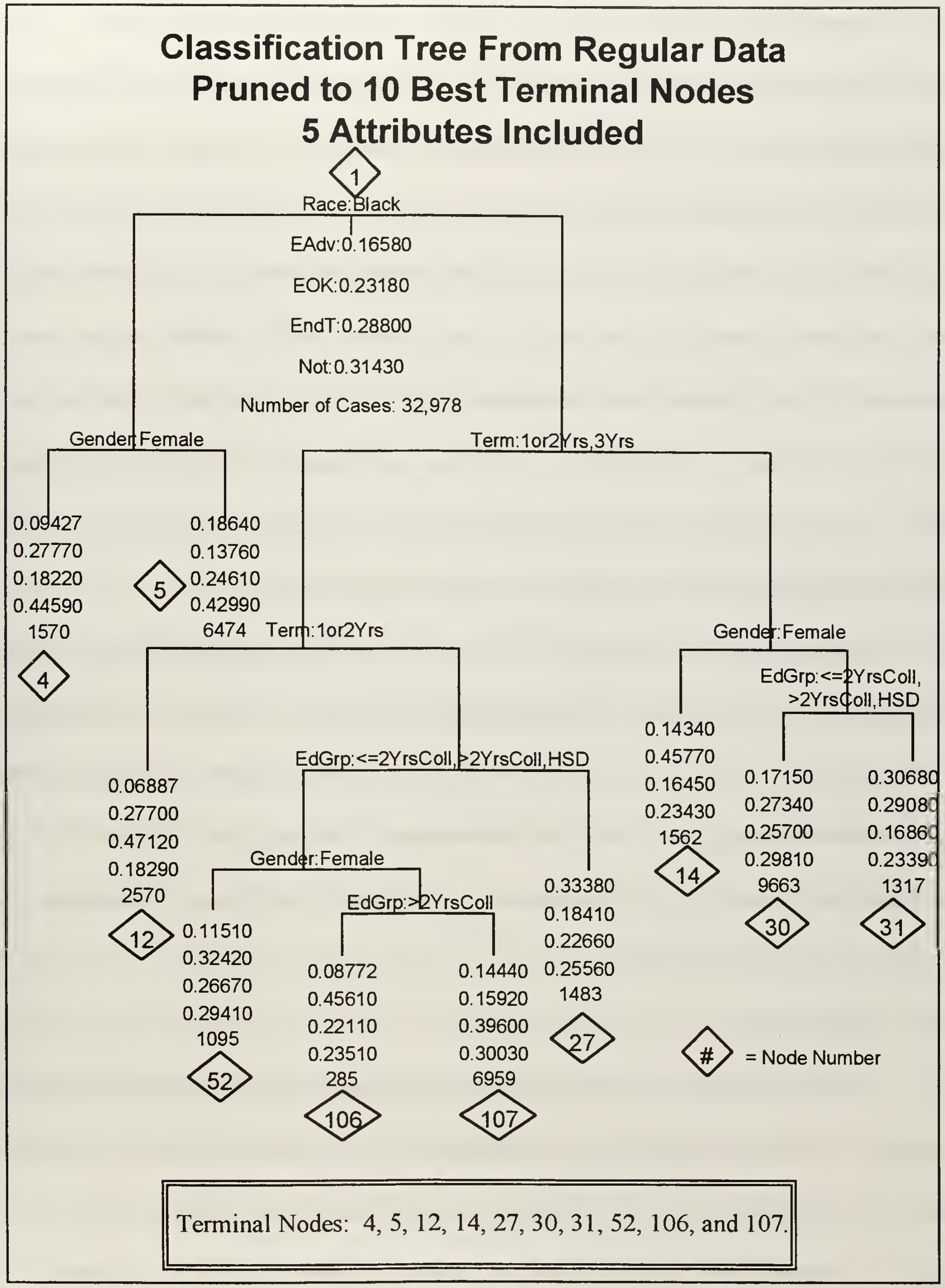


Figure 4.18 Pruned Classification Tree Using Regular Data With 5 Attributes

The terminal node with the largest number of cases in Figure 4.18 is node 30. This node contains 9,663 cases or approximately 29% of the total number of cases. The next largest node is node 107 with 21% and then node 5 with 20%. The three largest nodes are within ten percentage points of each other. This is the first time this has occurred. In the earlier analysis the largest nodes were all 32% or larger and the next largest node was more than ten percentage points away. The outcomes in the earlier analyzes were dominated by one large node and one smaller node. Why do three nodes dominate the outcome in this case? The format of the data is the main reason for having three dominant nodes. This is the first time that the analysis has been performed on the data with additional levels in the attributes AND with a fifth attribute present.

The large terminal nodes can be investigated and the Loss type categories can be examined by the same methods previously discussed. Figure 4.18 contains one last point to be discussed. While five attributes are provided as input, CART will only use the attributes necessary to grow a tree to a specified size. In this case, the attribute AFQT is not found in Figure 4.18. CART determined that splitting on AFQT was not necessary to grow a tree with 10 terminal nodes.

C. SUMMARY

All of the analyzes performed clearly demonstrated that including four Loss type categories was very beneficial. They provided much more information than just two. The cost of using four Loss types is an increase in the misclassification rate for the node. The purpose of the tree and how it is used will determine when additional Loss levels are desired and when a lower misclassification rate is preferred.

The attributes that characterize the cases in a terminal node are easily determined by tracing the path from the root node to the terminal node in question. One must know the number of cases in the terminal node in addition to the Loss type proportions to examine a terminal node. Investigating a terminal node by just looking at the proportions can be misleading. The number of cases in a terminal node will dictate which nodes to examine more closely. When a terminal node's attributes, the Loss type proportions and the number of cases in the terminal node are used together, they provide significant insight into the terminal node.

Using attributes with few levels results in terminal nodes with very broad characteristics. By increasing the levels of a particular attribute, the terminal nodes will be more tightly defined. This point was driven home in the analysis of the Regular data with 4 attributes. During this analysis, EdGrp was split on twice because its number of attributes was increased from two levels to five levels.

The CART process will determine which attributes are important and which are not important. Importance of an attribute is measured by how much that attribute contributes to reducing the deviance of the tree. When the attribute Race was added to the analysis, CART determined that it was the single most important attribute in reducing the deviance in the tree. Race became the dominant split attribute at the root node. The analysis of the Regular data with 5 attributes was an example of the other extreme. During this analysis, CART determined that even though the attribute AFQT was provided as input, it was not necessary to produce a tree with 10 terminal nodes.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

CART has been presented as a method of partitioning soldiers into groups that are more homogeneous, relative to their loss behavior, then other time series methods. Once CART's structure is understood, the method is less complicated to conduct and more easily understood than many other methods. The means of presenting the results from CART is highly visual. Reading and interpreting a tree can be quickly explained to an audience. An audience can readily understand how a tree flows from node to node. The simplicity in presentation and the ease of understanding are the greatest advantages of CART.

Deciding on which attributes to include when using CART is constrained only by the available computer power. In the initial assessment, if sufficient power is available, all relevant attributes should be included. CART will use only the attributes necessary to grow a tree to the desired size. The amount of information available used to make a decision can be severely limited when important variables are excluded. This point was clearly demonstrated in this thesis when analysis was performed on data first excluding the attribute Race and then including the attribute.

Once the attributes to use in CART are selected, the levels for each attribute must be established. Having too few levels of a particular attribute may lead to not having enough information available after the final tree has been created. Additional levels can lead to having terminal nodes that are more definitive in their characterization of the cases in the node. However, too many levels can over-define an attribute and can produce meaningless results. CART is useful in exploring which attribute levels should be selected.

CART can aid in identifying areas of concern. Suppose a large group of soldiers had a high rate of re-enlistment: they decided to stay in the Army past the end of their first-term of enlistment. It might be worth knowing what attributes characterize this group of soldiers. CART will determine the characteristics of this group and provide the proportions of those that stayed in the Army and those that did not. If structured correctly, CART will provide a breakdown by category of those soldiers who did not stay in the Army. The Loss type categories used in this thesis are an example of the breakdown CART can provide. Other categories or combinations of categories could be used in the analysis if desired.

B. RECOMMENDATIONS FOR FURTHER STUDY

Additional attributes should be added to the analysis to determine the combination that produces the greatest homogeneity in forecasting. Although the discussion of race can be a volatile subject, it has been shown that race as an attribute provides a great amount of predictive power. There are other attributes, and their levels, that should be explored to determine their importance. Two other attributes that should be investigated to determine their importance in forecasting are age and month of enlistment. It is important to point out that the data file that contains race, age, and month of enlistment is readily available. In fact, the data file that contains the attributes the Army currently uses (AFQT, Gender, EdGrp, and Term), also contains the attributes race, age, and month of enlistment. Extracting the additional attributes from the data file and using them in CART could result in a much greater predictive capability of the trees grown.

Analysis of the data should be performed after including new attributes and old attributes with new levels. The analysis could be used to explore the structure of the current C-Groups used in the Army. The hypothesis is that the current C-Groups no longer adequately describe the Army force structure. Should the C-Groups be changed? CART can provide valuable insights in answering this question. C-Groups must represent the current force structure in the Army and they must provide a high degree of predictability. CART can aid in determining the appropriate number of C-Groups and the structure of each group. Exploring various combinations of attributes and levels is an advantage CART has over other techniques.

This study was limited in scope due to the size of the original data file. The resources available were unable to handle the size of the original data file. Future studies should be performed with resources that are capable of handling the entire data file. When these studies are performed, the most recent data available should be used. The data available for this thesis included soldiers who entered the Army between January 1983 and December 1988. Additionally, new factors should be selected and appended to the data files so as to enhance the search for important explanatory variables.

Other administrative goals were ignored in this study and should be considered prior to conducting additional research. For example, separating soldiers by race could be an issue that is sensitive. While separating soldiers by gender can provide valuable insights into the force structure, gender can also be an issue that is sensitive. External issues may affect the formulation of the goals of future studies.

APPENDIX A. SUMMARY STATISTICS

		Entire Data Set	Sample File
AFQT			
	I	2%	2%
	II	34	35
	IIIA	27	28
	IIIB	30	30
	IV	7	5
	V	less than 0.1%	less than 0.1%
EdGrp			
	NHSD	9%	7%
	GED	4	4
	HSD	78	80
	<=2YrsColl	6	6
	>2YrsColl	3	3
Gender			
	Female	13%	14%
	Male	87	86
Term			
	1or2Yrs	8%	9%
	3Yrs	41	41
	4Yrs	49	48
	5or6Yrs	2	2
Race			
	White	72%	70%
	Black	23	24
	Other	5	6
Loss			
	EAdv	19%	17%
	EOK	23	23
	EndT	28	29
	Not	30	31

APPENDIX B. SAMPLE FILE (FIRST 40 ROWS)

```
   AFQT Gender      EdGrp    Term  Race Loss
 1   IV   Male        HSD    3Yrs White EndT
 2   II   Male        HSD    4Yrs White EAdv
 3   II   Male        GED    3Yrs White  Not
 4 IIIB   Male        HSD    3Yrs White EndT
 5   II   Male        HSD    4Yrs White EndT
 6   IV   Male        HSD    3Yrs White  Not
 7   II Female        HSD    4Yrs White  EOK
 8   II   Male        HSD    4Yrs White  Not
 9 IIIB   Male        HSD    4Yrs White EndT
10 IIIB   Male        HSD    4Yrs White  EOK
11   II   Male        HSD    4Yrs White EAdv
12 IIIA   Male      NoHSD    3Yrs White  Not
13 IIIA   Male        HSD 1or2Yrs White EndT
14 IIIA   Male        HSD    4Yrs White EAdv
15   II   Male        HSD    3Yrs White EndT
16   II   Male      NoHSD    3Yrs White  EOK
17 IIIA Female        HSD    4Yrs White  EOK
18 IIIB   Male        HSD    4Yrs White  Not
19 IIIA   Male        GED    3Yrs White EAdv
20 IIIB Female        HSD    3Yrs White  EOK
21 IIIA   Male        HSD    3Yrs White EndT
22 IIIA   Male        HSD    4Yrs Black EndT
23 IIIB   Male        HSD    3Yrs Black  Not
24 IIIA   Male      NoHSD    3Yrs White EAdv
25 IIIB   Male >2YrsColl    3Yrs White EndT
26 IIIB   Male        HSD    3Yrs White EAdv
27 IIIA   Male        HSD    4Yrs White EndT
28 IIIB   Male        HSD    3Yrs White  Not
29 IIIA   Male         HSD    4Yrs White EndT
30   II   Male         HSD    3Yrs Black  EOK
31 IIIB   Male         HSD    3Yrs Black EAdv
32 IIIB   Male         HSD    3Yrs White  Not
33   II Female         HSD    3Yrs White EAdv
34   II   Male <=2YrsColl    3Yrs White  Not
35 IIIA   Male         HSD    3Yrs White  EOK
36 IIIB   Male         HSD    3Yrs Other  Not
37   II   Male         HSD    4Yrs White EndT
38 IIIB   Male <=2YrsColl    3Yrs White  Not
39 IIIA   Male         GED    3Yrs White EndT
40 IIIA Female         HSD    4Yrs White EndT
```

APPENDIX C. CROSS VALIDATION METHOD IN S-PLUS

The purpose of this appendix is to provide details on how to use the cross-validation (CV) method with the stopping criteria removed. When the data file is small, as it is in the example in Chapter III, it is necessary to create the initial overgrown tree with the stopping criteria removed. If the default stopping criteria are left in place, the tree will not be large enough to uncover the entire structure of the data. In order to grow a tree with the stopping criteria removed, the inputs to the tree method must include the following:

```
minsize=2
mindev=0
```

The problem arises when a tree object is grown using the tree method with the stopping criteria removed and then this tree object is used as the input to the CV method. The tree method is used within the CV method but the default stopping criteria are left in place. To override the default values, one must actually adjust the code of the CV method.

The cross-validation (CV) method can take several parameters as inputs. At a minimum, a tree object must be provided. The reminder of the parameters are optional. If certain parameters are not provided, the CV method will provide a default. As used in this document, a tree object and the pruning method were provided to the the CV method. No other inputs were provided. The following is the code for the CV method as it is used in this document with the stopping criteria in place. The lines have been numbered in order to reference them.

```
1       cv.tree
2       function(object=tree object, rand, FUN = prune.tree, ..., big = F)
3       {
4               if(!inherits(object, "tree"))
5                       stop("Not legitimate tree")
6               m <- model.frame(object)
7               call <- match.call()
8               method <- call$method
9               p <- FUN(object, ...)
10              if(missing(rand))
11                      rand <- sample(10, length(m[[1]]), replace = T)
12              which <- unique(rand)
13              cvdev <- 0
14              pk <- p$k
15              expr <- expression({
16                      tlearn <- tree(model = .m[.rand != i,  , drop = F])
17                  plearn <- .FUN(tlearn, newdata = .m[.rand == i,  , drop = F],
18                                                                  k = .pk)
19                  .cvdev <- .cvdev + plearn$dev
20          }
21          )[[1]]
22          if(!is.null(method))
23                  expr[[2]][[2]]$method <- eval(method, sys.parent())
24          if(!big)
25                  for(i in which) {
26                          tlearn <- tree(model = m[rand != i,  , drop = F])
27                          plearn <- FUN(tlearn, newdata = m[rand == i,  , drop = F],
28                                                                        k = pk)
29                          cvdev <- cvdev + plearn$dev
30                  }
31          else {
32                  assign(".m", m, w = 1)
33                  assign(".FUN", FUN, w = 1)
34                  assign(".rand", rand, w = 1)
35                  assign(".cvdev", cvdev, w = 1)
36                  assign(".pk", pk, w = 1)
37                  eval(substitute(For(i = unique(.rand), expr)), list(expr = expr))
38                  cvdev <- get(".cvdev", w = 1)
39                  remove(c(".m", ".FUN", ".rand", ".cvdev", ".pk"), w = 1)
40          }
41          p$dev <- cvdev
42          p
43  }
```

Here is the author's procedure for using the CV method with the stopping criteria removed.

* Make a copy cv.tree. Call it my.cv.tree. The command in S-Plus is:
  > my.cv.tree_cv.tree
* The code in my.cv.tree must be changed. This is done by "fixing" the file.
  The S-Plus command is: > fix(my.cv.tree)
  This will open up Notepad with the code.
* Lines 16 and 26 must be changed to include: minsize=2 and mindev=0.
  When the changes are complete, the lines will look as follows:

line 16: tlearn <- tree(model = .m[.rand != i, , drop = F],minsize=2,mindev=0)
line 26: tlearn <- tree(model = m[rand != i, , drop = F],minsize=2,mindev=0)

* Save the new file in Notepad. DO NOT PROVIDE A NAME. Exit Notepad.
* Use my.cv.tree just as you would cv.tree.

APPENDIX D. EXAMPLE DATA AND S-PLUS COMMANDS

The following is the data used in the CART example found in Chapter III. The file in S-Plus was called “sub.samp.”

```
>sub.samp
   AFQT Gender EdGrp Term  Race    Loss
 1   II   Male   HSD 3Yrs Other    lost
 2 IIIB   Male   HSD 3Yrs White    lost
 3 IIIA Female   HSD 4Yrs Black    lost
 4 IIIB Female   HSD 4Yrs Other    lost
 5 IIIA   Male   HSD 4Yrs White    lost
 6 IIIA   Male   HSD 2Yrs White    lost
 7    I   Male   HSD 2Yrs White    lost
 8   IV   Male   HSD 5Yrs Black    lost
 9   II   Male NoHSD 4Yrs White notlost
10   II   Male NoHSD 5Yrs White    lost
11   II   Male   HSD 4Yrs White    lost
12   II   Male   HSD 4Yrs White    lost
13 IIIB   Male   HSD 3Yrs Black notlost
14 IIIB Female   HSD 4Yrs Black notlost
15   II   Male   HSD 3Yrs White    lost
16   II   Male   HSD 4Yrs White    lost
17   II   Male   HSD 3Yrs White notlost
18 IIIA   Male   HSD 4Yrs White    lost
19   II   Male   HSD 4Yrs White    lost
20    I   Male NoHSD 3Yrs White notlost
21   IV   Male   HSD 3Yrs Other    lost
22 IIIA   Male   HSD 3Yrs White    lost
23 IIIB   Male   HSD 3Yrs White notlost
24 IIIA   Male   HSD 4Yrs Black    lost
25   II   Male NoHSD 4Yrs White    lost
26 IIIB   Male   HSD 4Yrs White    lost
27 IIIB   Male   HSD 3Yrs White    lost
28 IIIB   Male   HSD 6Yrs Black    lost
29 IIIB Female   HSD 4Yrs White notlost
30 IIIB   Male   HSD 4Yrs Black    lost
31 IIIB   Male   HSD 3Yrs White notlost
32   II   Male   HSD 4Yrs Other notlost
33   II   Male   HSD 3Yrs Black notlost
34 IIIA   Male   HSD 4Yrs White    lost
35   II   Male   HSD 4Yrs Other    lost
36   IV   Male   HSD 3Yrs Black    lost
37 IIIB   Male   HSD 3Yrs Black    lost
38    I   Male   HSD 3Yrs White notlost
39 IIIB   Male   HSD 3Yrs Black    lost
40   II   Male   HSD 2Yrs White    lost
41 IIIB   Male   HSD 4Yrs White    lost
42   II Female   HSD 3Yrs Other    lost
43   II Female   HSD 4Yrs White    lost
44 IIIA   Male NoHSD 4Yrs White    lost
```

```
AFQT Gender EdGrp Term  Race    Loss
45 IIIB Female   HSD 3Yrs White    lost
46 IIIB   Male   HSD 4Yrs White notlost
47 IIIB   Male   HSD 4Yrs Other notlost
48 IIIA   Male   HSD 4Yrs Other notlost
49 IIIB Female   HSD 4Yrs White    lost
50   II   Male   HSD 2Yrs White    lost
```

S-Plus commands used for the example in Chapter III are as follows:

```
> sub.tree_tree(Loss~Gender+EdGrp+AFQT+Term,data=sub.samp,
+ minsize=2,mindev=0)
> plot(sub.tree)                              #Figure 3.4
> text(sub.tree,label="yprob",pretty=0,all=F)
> title(main=”Classification Tree For Example Data”)
> sub.prune_prune.tree(sub.tree)
> plot(sub.prune)                             #Figure 3.5
> sub.cv_my.cv.tree(sub.tree,FUN=prune.tree)        # See Appendix C
> plot(sub.cv)                                #Figure 3.6
> sub.best_prune.tree(sub.tree,best=4)
> plot(sub.best)                              #Figure 3.3
> text(sub.best,label=”yprob”,pretty=0,all=T)
```

NOTE: All plots created in S-Plus were copied into Power Point and adjusted prior to their inclusion in this document.

APPENDIX E. C*-GROUP DATA AND S-PLUS COMMANDS

C*-Group data was used in Chapter IV. The following is the first 40 rows of the C*-Group data. The file name in S-Plus was “cgroup.data.”

```
> cgroup.data[1:40,]
     AFQT Gender EdGrp   Term  Race Loss
 1   IV-V   Male   HSD 3&4Yrs White EndT
 2 I-IIIA   Male   HSD 3&4Yrs White EAdv
 3 I-IIIA   Male NoHSD 3&4Yrs White  Not
 4   IIIB   Male   HSD 3&4Yrs White EndT
 5 I-IIIA   Male   HSD 3&4Yrs White EndT
 6   IV-V   Male   HSD 3&4Yrs White  Not
 7 I-IIIA Female   HSD 3&4Yrs White  EOK
 8 I-IIIA   Male   HSD 3&4Yrs White  Not
 9   IIIB   Male   HSD 3&4Yrs White EndT
10   IIIB   Male   HSD 3&4Yrs White  EOK
11 I-IIIA   Male   HSD 3&4Yrs White EAdv
12 I-IIIA   Male NoHSD 3&4Yrs White  Not
13 I-IIIA   Male   HSD  Other White EndT
14 I-IIIA   Male   HSD 3&4Yrs White EAdv
15 I-IIIA   Male   HSD 3&4Yrs White EndT
16 I-IIIA   Male NoHSD 3&4Yrs White  EOK
17 I-IIIA Female   HSD 3&4Yrs White  EOK
18   IIIB   Male   HSD 3&4Yrs White  Not
19 I-IIIA   Male NoHSD 3&4Yrs White EAdv
20   IIIB Female   HSD 3&4Yrs White  EOK
21 I-IIIA   Male   HSD 3&4Yrs White EndT
22 I-IIIA   Male   HSD 3&4Yrs Black EndT
23   IIIB   Male   HSD 3&4Yrs Black  Not
24 I-IIIA   Male NoHSD 3&4Yrs White EAdv
25   IIIB   Male   HSD 3&4Yrs White EndT
26   IIIB   Male   HSD 3&4Yrs White EAdv
27 I-IIIA   Male   HSD 3&4Yrs White EndT
28   IIIB   Male   HSD 3&4Yrs White  Not
29 I-IIIA   Male   HSD 3&4Yrs White EndT
30 I-IIIA   Male   HSD 3&4Yrs Black  EOK
31   IIIB   Male   HSD 3&4Yrs Black EAdv
32   IIIB   Male   HSD 3&4Yrs White  Not
33 I-IIIA Female   HSD 3&4Yrs White EAdv
34 I-IIIA   Male   HSD 3&4Yrs White  Not
35 I-IIIA   Male   HSD 3&4Yrs White  EOK
36   IIIB   Male   HSD 3&4Yrs Other  Not
37 I-IIIA   Male   HSD 3&4Yrs White EndT
38   IIIB   Male   HSD 3&4Yrs White  Not
39 I-IIIA   Male NoHSD 3&4Yrs White EndT
40 I-IIIA Female   HSD 3&4Yrs White EndT
```

S-Plus commands used in Chapter IV during the analysis of the C*-Group data are as follows:

```
> # Create overgrown tree from C-Group data only using 4 attributes.  Plot tree.
> cgroup4.tree_tree(Loss~AFQT+Gender+EdGrp+Term,data=cgroup.data)
> plot(cgroup4.tree)                          # Figure 4.1
> text(cgroup4.tree,pretty=0,label="yprob",all=T)
> summary(cgroup4.tree)
Classification tree:
tree(formula = Loss ~ AFQT + Gender + EdGrp + Term, data = cgroup.data)
Number of terminal nodes:  16
Residual mean deviance:  2.665 = 87830 / 32960
Misclassification error rate: 0.6512 = 21475 / 32978

> # Execute pruning and cross-validation methods, and plot.
> cgroup4.prune_prune.tree(cgroup4.tree)
> cgroup4.cv_cv.tree(cgroup4.tree,FUN=prune.tree)
> plot(cgroup4.prune)                         # Figure 4.2
> plot(cgroup4.cv)                            # Figure 4.3

> # Prune cgroup4.tree to 10 best terminal nodes.  Plot tree.
> cgroup4.best10_prune.tree(cgroup4.tree,best=10)
> plot(cgroup4.best10)                        # Figure 4.4
> text(cgroup4.best10,label="yprob",all=T,pretty=0)
> summary(cgroup4.best10)
Classification tree:
snip.tree(tree = cgroup4.tree, nodes = c(29, 4, 5))
Number of terminal nodes:  10
Residual mean deviance:  2.665 = 87850 / 32970
Misclassification error rate: 0.6513 = 21478 / 32978

> # Create overgrown tree from C-Group data only using 5 attributes.  Plot tree.
> cgroup.tree_tree(Loss~AFQT+Gender+EdGrp+Term+Race,data=cgroup.data)
> plot(cgroup.tree)                           # Figure 4.5
> text(cgroup.tree,pretty=0,label="yprob",all=T)
> summary(cgroup.tree)
Classification tree:
tree(formula = Loss ~ AFQT + Gender + EdGrp + Term + Race, data = cgroup.data)
Number of terminal nodes:  35
Residual mean deviance:  2.643 = 87050 / 32940
Misclassification error rate: 0.6334 = 20887 / 32978
```

```
> # Execute pruning and cross-validation methods, and plot.
> cgroup.prune_prune.tree(cgroup.tree)
> cgroup.cv_cv.tree(cgroup.tree,FUN=prune.tree)
> plot(cgroup.prune)                              # Figure 4.6
> plot(cgroup.cv)                                 # Figure 4.7

> # Prune cgroup.tree to 9 best terminal nodes.  Plot tree.
> cgroup.best9_prune.tree(cgroup.tree,best=9)
> plot(cgroup.best9)                              # Figure 4.8
> text(cgroup.best9,label="yprob",all=T,pretty=0)
> summary(cgroup.best9)
Classification tree:
snip.tree(tree = cgroup.tree, nodes = c(20, 4, 53, 52, 11, 27, 12, 21, 7))
Number of terminal nodes:  9
Residual mean deviance:  2.649 = 87350 / 32970
Misclassification error rate: 0.6387 = 21062 / 32978
```

NOTE: All plots created in S-Plus were copied into Power Point and adjusted prior to their inclusion in this document.

APPENDIX F. S-PLUS COMMANDS USED ON THE REGULAR DATA

Regular data was used in Chapter IV. The first 40 rows of the Regular data are available Appendix B. The file name in S-Plus was “samp.file.”

S-Plus commands used in Chapter IV during the analysis of the Regular data are as follows:

```
> # Create overgrown tree from Regular data only using 4 attributes.  Plot tree.
> samp4.tree_tree(Loss~AFQT+Gender+EdGrp+Term,data=samp.file)
> plot(samp4.tree)                              # Figure 4.9
> text(samp4.tree)
> summary(samp4.tree)
Classification tree:
tree(formula = Loss ~ AFQT + Gender + EdGrp + Term, data = samp.file)
Number of terminal nodes:  68
Residual mean deviance:  2.621 = 86260 / 32910
Misclassification error rate: 0.6294 = 20757 / 32978

> # Execute pruning and cross-validation methods, and plot.
> samp4.prune_prune.tree(samp4.tree)
> samp4.cv_cv.tree(samp4.tree,FUN=prune.tree)
> plot(samp4.prune)                             # Figure 4.10
> plot(samp.cv)                                 # Figure 4.11

> # Prune samp4.tree to 15 best terminal nodes.  Plot tree.
> samp4.best15_prune.tree(samp4.tree, best=15)
> plot(samp4.best15)                            # Figure 4.12
> text(samp4.best15,label="yprob",all=T,pretty=0)
> summary(samp4.best15)
Classification tree:
snip.tree(tree = samp4.tree, nodes = c(84, 85, 59, 8, 6, 15, 20, 9, 58, 23, 57, 87))
Number of terminal nodes:  15
Residual mean deviance:  2.631 = 86720 / 32960
Misclassification error rate: 0.6331 = 20877 / 32978

> # Prune samp4.tree to 10 best terminal nodes.  Plot tree.
> samp4.best10_prune.tree(samp4.tree, best=10)
> plot(samp4.best10)                            # Figure 4.13
> text(samp4.best10,label="yprob",all=T,pretty=0)
> summary(samp4.best10)
Classification tree:
snip.tree(tree = samp4.tree, nodes = c(6, 15, 20, 87, 28, 11, 29, 42, 4))
Number of terminal nodes:  10
Residual mean deviance:  2.635 = 86890 / 32970
Misclassification error rate: 0.6374 = 21019 / 32978

> # Create overgrown tree from regular data using 5 attributes.  Plot tree.
> samp.tree_tree(Loss~AFQT+Gender+EdGrp+Term+Race,data=samp.file)
> plot(samp.tree)                              # Figure 4.14
> text(samp.tree,pretty=0)
> summary(samp.tree)
Classification tree:
tree(formula = Loss ~ AFQT + Gender + EdGrp + Term + Race, data = samp.file)
Number of terminal nodes:  104
Residual mean deviance:  2.601 = 85500 / 32870
Misclassification error rate: 0.6161 = 20318 / 32978

> # Execute pruning and cross-validation methods, and plot.
> samp.prune_prune.tree(samp.tree)
> samp.cv_cv.tree(samp.tree,FUN=prune.tree)
> plot(samp.prune)                             # Figure 4.15
> plot(samp.cv)                                # Figure 4.16

> # Prune samp.tree to 15 best terminal nodes.  Plot tree.
> samp.best15_prune.tree(samp.tree, best=15)
> plot(samp.best15)                            # Figure 4.17
> text(samp.best15,label="yprob",all=T,pretty=0)
> summary(samp.best15)
Classification tree:
snip.tree(tree = samp.tree, nodes = c(20, 31, 11, 52, 215, 27, 14, 60, 4, 214, 61, 42, 12))
Number of terminal nodes:  15
Residual mean deviance:  2.615 = 86210 / 32960
Misclassification error rate: 0.6231 = 20548 / 32978

> # Prune samp.tree to 10 best terminal nodes.  Plot tree.
> samp.best10_prune.tree(samp.tree, best=10)
> plot(samp.best10)                            # Figure 4.18
> text(samp.best10,label="yprob",all=T,pretty=0)
> summary(samp.best10)
Classification tree:
snip.tree(tree = samp.tree, nodes = c(31, 52, 27, 14, 4, 12, 107, 30, 5))
Variables actually used in tree construction:
[1] "Race"   "Gender" "Term"   "EdGrp"
Number of terminal nodes:  10
Residual mean deviance:  2.627 = 86600 / 32970
Misclassification error rate: 0.6231 = 20548 / 32978
```

NOTE: All plots created in S-Plus were copied into Power Point and adjusted prior to their inclusion in this document.

LIST OF REFERENCES

Bartholomew, D.J., Forbes, and McClean, S.I., *Statistical Techniques for Manpower Planning*, Second Edition, John Wiley & Sons, 1991.

Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J., *Classification and Regression Trees*, Wadsworth International Group, 1984.

Chambers, J.M. and Hastie, T.J., *Statistical Models in S*, Wadsworth & Brooks/Cole Advanced Books and Software.

DeWald, E.T., *A Time Series Analysis of U.S. Army Enlisted Force Loss Rates*, MS Thesis, Operations Research Department, Naval Postgraduate School, 1996.

GRC, *ADP Support Services For Enlisted System: ELIM-COMPLIP System Specification*, General Research Corporation, McLean, Virginia, 1989.

Weiss, M.A., *Data Structures and Algorithm Analysis*, Second Edition, The Benjamin/Cummings Publishing Company, Inc., 1995.

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center 2
8725 John J. Kingman Rd., STE 0944
Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Rd.
Monterey, CA 93943-5101

3. Chief, Personnel Forecasting 2
ODCSPER, Rm 2C744
300 Army Pentagon
Washington, DC 20310-0300

4. CDR Timothy French, USN, Code 4111 1
Naval Supply Systems Command
5450 Carlisle Pike
P.O. box 2050
Mechanicsburg, PA 17055-0791

5. Mr. Jere Engelman, Code M041 1
NAVICP-Mechanicsburg, Code M041
5450 Carlisle Pike
Mechanicsburg, PA 17055-0788

6. Professor Robert R. Read 2
Operations Research Department, Code OR/Re
Naval Postgraduate School
Monterey, CA 93943

7. Professor Samuel E. Buttrey 1
Operations Research Department, Code OR
Naval Postgraduate School
Monterey, California 93943

8. LCDR Terence S. Purcell, USN 2
149 Progress Place
Reedsville, Pennsylvania 17084

UX LIBRARY
POSTGRADUATE SCHOOL
CA 93943-5101

DUDLEY KNOX LIBRARY

3 2768 003[illegible]9099 8